

PHP必须知道的300个问题

快速服务：微博、QQ在线服务

自学视频：40集大型多媒体自学视频

海量资源：模块库、案例库、素材库、题库

潘凯华　李　慧　刘　欣　等编著

本书提供了内容丰富的配套资源，可以登录www.tup.com.cn，找到本书后，在该页面的“网络资源”超链接处下载。也可以访问本书的新浪微博，根据提示链接下载。

清华大学出版社

PHP 必须知道的 300 个问题

潘凯华　李　慧　刘　欣　等编著

（模块库、案例库、素材库、题库）

（微博、QQ、论坛技术支持）

清华大学出版社

北　京

内 容 简 介

本书以PHP基础知识结构为框架，讲解各部分知识所对应的常见开发疑难问题，并进行透彻地解析。内容包括PHP开发规范与入门要点、PHP基础、Web技术、数据库操作、日期和时间处理、图形图像处理、文件与目录操作、面向对象的程序设计、数据库抽象层、Ajax技术、XML技术、PDF技术、PHP与多媒体技术、Smarty模板、ThinkPHP框架和PHP项目开发技巧。

本书所涉及的疑难问题都是一线开发人员在实际项目中经常遇到的，针对这些问题给出了解决方案，并进行了技术上的解析。读者通过对本书的学习，能够提高实际解决问题的能力。

本书提供了源程序、素材，提供了相关的模块库、案例库、素材库、题库等多种形式的辅助学习资料，还提供迅速及时的微博、QQ、论坛等技术支持。

本书内容详尽，实例丰富，非常适合作为零基础学习人员的学习用书和大中专院校师生的学习教材，也适合作为相关培训机构的师生和软件开发人员的参考资料。

图书在版编目（CIP）数据

PHP必须知道的300个问题/潘凯华，李慧，刘欣等编著．—北京：清华大学出版社，2012.2

ISBN 978-7-302-27204-5

I. ①P…　II. ①潘…　②李…　③刘…　III. ①PHP语言-程序设计　IV. ①TP312

中国版本图书馆CIP数据核字（2011）第223040号

责任编辑：赵洛育　刘利民
版式设计：文森时代
责任校对：张兴旺
责任印制：
出版发行：清华大学出版社
　　网　　址：http://www.tup.com.cn，http://www.wqbook.com
　　地　　址：北京清华大学学研大厦A座　　　**邮　　编：**100084
　　社 总 机：010-62770175　　　**邮　　购：**010-62786544
　　投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
　　质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 刷 者：
装 订 者：
经　　销：全国新华书店
开　　本：185mm×260mm　**印　　张：**32.25　**字　　数：**931千字
版　　次：2012年2月第1版　**印　　次：**2012年2月第1次印刷
印　　数：1～5000
定　　价：59.80元

产品编号：043922-01

前　言

Preface

程序员在求职时，经常会被问到有关开发经验的问题，例如“从事了多少年的开发”、“参与了哪些大型项目”等。为何面试官对项目经验丰富的求职者青睐有加？究其根源在于程序开发是一门实践性非常强的学科，正所谓“纸上得来终觉浅，绝知此事要躬行”。编程经验是程序开发者在实践中积累的技术资源库，其中包括各种习惯用法、知识理论、代码片段、应用技巧等。而编程高手一般也是站在前人的肩膀上，分析别人的代码而有所领悟，或者借鉴他人的经验技巧形成自己的技能，在认识错误与解决问题中进步。这也是每个编程者的必由之路。

本书汇集了一线开发人员常见的 300 个 PHP 编程疑难问题，每个问题都给出了详细的解答与点评，图文并茂，难易并举，通过本书的学习，相信读者可以尽享前人的开发经验，迅速提升个人的开发能力。

本书内容

本书以 PHP 基础知识结构为框架，给出了每部分知识中可能遇到的疑难问题或开发技巧。全书共分为 16 章，具体包括：PHP 开发规范与入门要点、PHP 基础、Web 技术、数据库操作、日期和时间处理、图形图像处理、文件与目录操作、面向对象的程序设计、数据库抽象层、Ajax 技术、XML 技术、PDF 技术、PHP 与多媒体技术、Smarty 模板、ThinkPHP 框架和 PHP 项目开发技巧。

为了更清晰地阐述问题和给出问题的解决方案，本书设置了以下栏目。

☑ **问题阐述**

对问题进行详细阐述，对复杂问题给出问题产生的条件，并对问题中需要解决的部分做出说明。

☑ **专家解答**

根据具体问题进行具体分析，给出问题的正确答案及解题思路，并提供该问题涉及的技术知识。

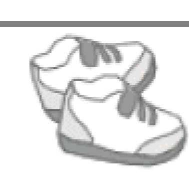
☑ **专家点评**

对问题及解答做出总结，为以后再出现类似问题提供解决思路，并给出此问题需要注意的事项和对该问题的一些拓展。

本书特色

☑ **贴近应用**

本书精选的疑难问题都是在实际项目开发中经常遇到的问题，可以解决读者在编程中可能





遇到的疑惑。

☑ **横向链接**

本书知识框架与《PHP 快速入门及项目实战》一书相对应，读者可以在使用《PHP 快速入门及项目实战》一书进行基础学习之后，再使用本书丰富并提高技能。

☑ **解析透彻**

本书对每个问题的相关知识进行细致地讲解，并进行知识拓展，使读者不仅知其然，而且知其所以然。

☑ **授人以渔**

本书在讲解技术的同时还注重对读者能力的培养，帮助读者提高分析问题与解决问题的能力。

本书配套资源

本书提供了内容丰富的配套资源包括：源程序、素材，以及模块库、案例库、题库、素材库等多项辅助内容，读者朋友可以通过如下方式获取。

第 1 种方式：

（1）登录 www.tup.com.cn，在网页右上角的搜索文本框中输入本书书名（注意区分大小写和留出空格），或者输入本书关键字，或者输入本书 ISBN 号（注意去掉 ISBN 号间隔线“-”），单击“搜索”按钮。

（2）找到本书后单击超链接，在该书的网页下侧单击“网络资源”超链接，即可下载。

第 2 种方式：

访问本书的新浪微博 PHPbook，找到配套资源的链接地址进行下载。

读者人群

本书非常适合以下人员阅读：

☑ 从事 PHP 语言编程行业的开发人员

☑ 有一定语言基础，想进一步提高技能的人员

☑ 大中专院校的老师和学生

☑ 即将走上工作岗位的大学毕业生

☑ 相关培训机构的老师和学员

☑ PHP 编程爱好者

读者服务&本书勘误

读者在使用本书过程中遇到的所有问题，均可通过以下方式联系我们。

1．新浪微博：PHPbook。

及时发布读者答疑、本书勘误、配套资料更新等内容。

2．腾讯 QQ：4006751066。

3．登录网站：www.mingribook.com，在论坛、勘误发布、读者纠错、技术支持、读者之家等栏目中的相关模块中提问、留言或查看。





本书作者

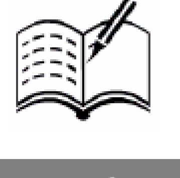
本书由明日科技组织编写，参加编写的有潘凯华、李慧、刘欣、王小科、赵会东、聂喜婷、宋环雨、李继业、赛奎春、杨丽、高春艳、陈丹丹、王国辉、张振坤、李伟、沈博、孙秀梅、曹飞飞、王雪、朱晓、赵永发、刘龄龄、孙茜、陈英、肖鑫和刘冠男等。由于作者水平有限，疏漏和不足之处在所难免，敬请广大读者朋友批评指正。

编　者

目　录

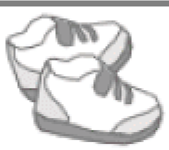
Contents




Note

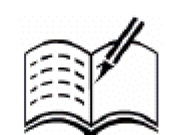


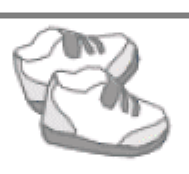

Note

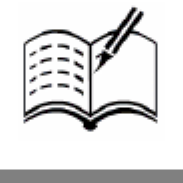



Note



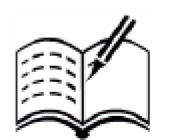


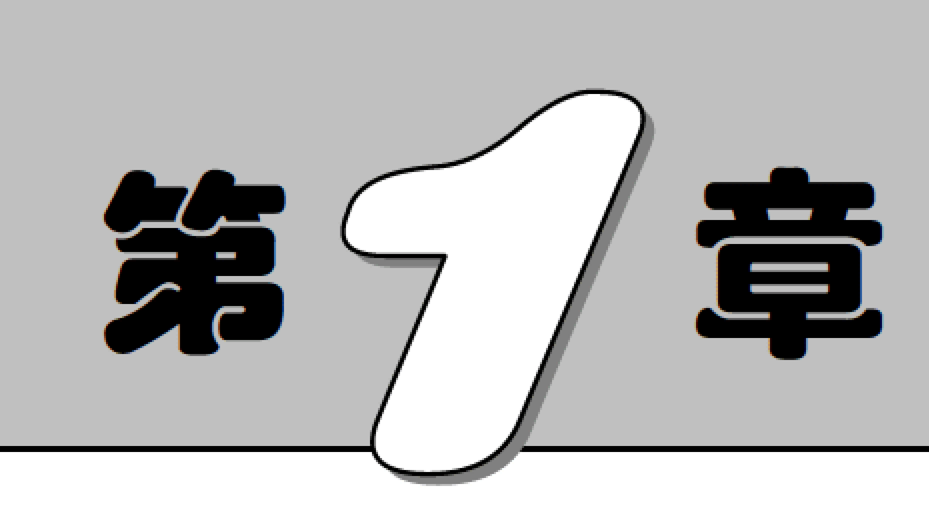




Note

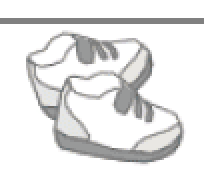



Note

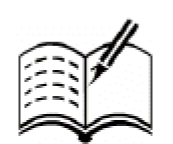


PHP 开发规范与入门要点

- 问题 1　如何在 Windows 下配置 PHP 开发环境?
- 问题 2　如何在 Linux 下配置 PHP 开发环境?
- 问题 3　如何搭建 IIS+PHP+MySQL 环境?
- 问题 4　PHP 集成开发环境的特点有哪些?
- 问题 5　如何应用 AppServ 搭建 PHP 开发环境?
- 问题 6　如何通过 XAMPP 配置 PHP 开发环境?
- 问题 7　XAMPP——Linux 版 PHP 集成化安装包
- 问题 8　Apache 配置文件全解
- 问题 9　PHP.INI 配置文件全解
- 问题 10　Apache 不能处理 PHP 文件
- 问题 11　无法找到 httpd.h 文件
- 问题 12　Apache 不能启动的解决方法
- 问题 13　如何解决 Apache 服务器端口冲突?
- 问题 14　如何配置 Apache 虚拟主机?
- 问题 15　如何动态加载扩展库?
- 问题 16　register_globals（注册为全局变量）的开启与关闭
- 问题 17　为什么要设置时区?
- 问题 18　如何在 Dreamweaver 中创建站点?
- 问题 19　如何使用 Dreamweaver 开发 PHP 程序?
- 问题 20　如何使用 Dreamweaver 识别.phtml 的模板文件?
- 问题 21　如何在 Windows 下安装和配置 phpMyAdmin?
- 问题 22　如何在 Linux 下安装和配置 phpMyAdmin?
- 问题 23　如何应用 phpMyAdmin 修改数据库用户名和密码?
- 问题 24　如何应用 phpMyAdmin 创建数据库与数据表?
- 问题 25　如何应用 phpMyAdmin 导入和导出 MySQL 数据库脚本?
- 问题 26　如何打开 magic_quotes_gpc 来防止 SQL 注入?
- 问题 27　如何对错误提示信息进行控制?
- 问题 28　如何记录错误日志?
- 问题 29　如何解决使用 MySQL 函数导致 Apache(IIS)崩溃的问题?
- 问题 30　HTTP Apache 错误代码都有哪些? 分别代表什么含义?



问题 1　如何在 Windows 下配置 PHP 开发环境？



Note

问题阐述

PHP 高效、稳定的运行依赖于服务器的编译和执行，那么如何在微软的 Windows 操作系统中架设安全、可靠的 PHP 运行环境呢？

专家解答

1. Windows 下 Apache 的安装配置

Apache 是全世界范围内使用最广的 Web 服务软件。超过 50%的网站都在使用 Apache，它以高效、稳定、安全、免费（最重要的一点）的优点而成为最受欢迎的服务器软件。

本节主要介绍如何在 Windows 操作系统中安装和配置 Apache 服务器。在安装 Apache 服务器前，应到其官方网站 http://www.apache.org 下载 Apache 的安装程序。

在 Windows 下实现 Apache 的安装的操作步骤如下。

（1）下载 Apache 的安装包“apache_2.0.59-win32-x86-no_ssl.msi”后双击该安装包，打开如图 1.1 所示的 Apache 的安装窗口。

（2）在图 1.1 所示的窗口中单击 Next 按钮，打开如图 1.2 所示的 Apache 许可协议窗口，选中图中的 I accept the terms in the license agreement 单选按钮，然后单击 Next 按钮。

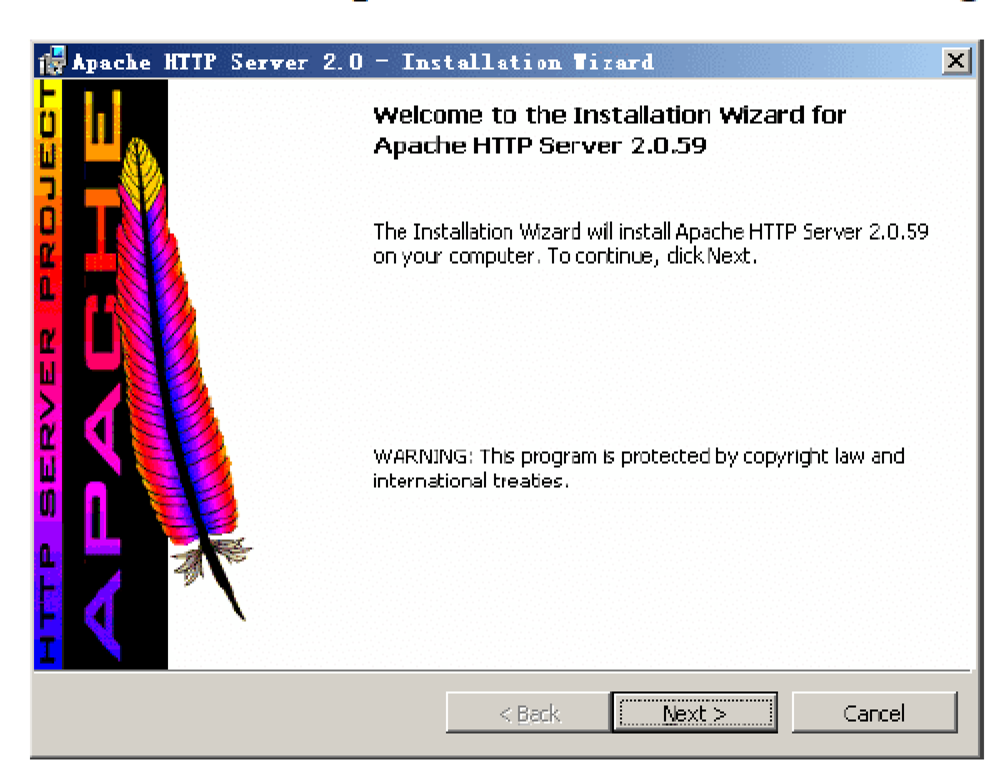


图 1.1　Apache 的安装窗口

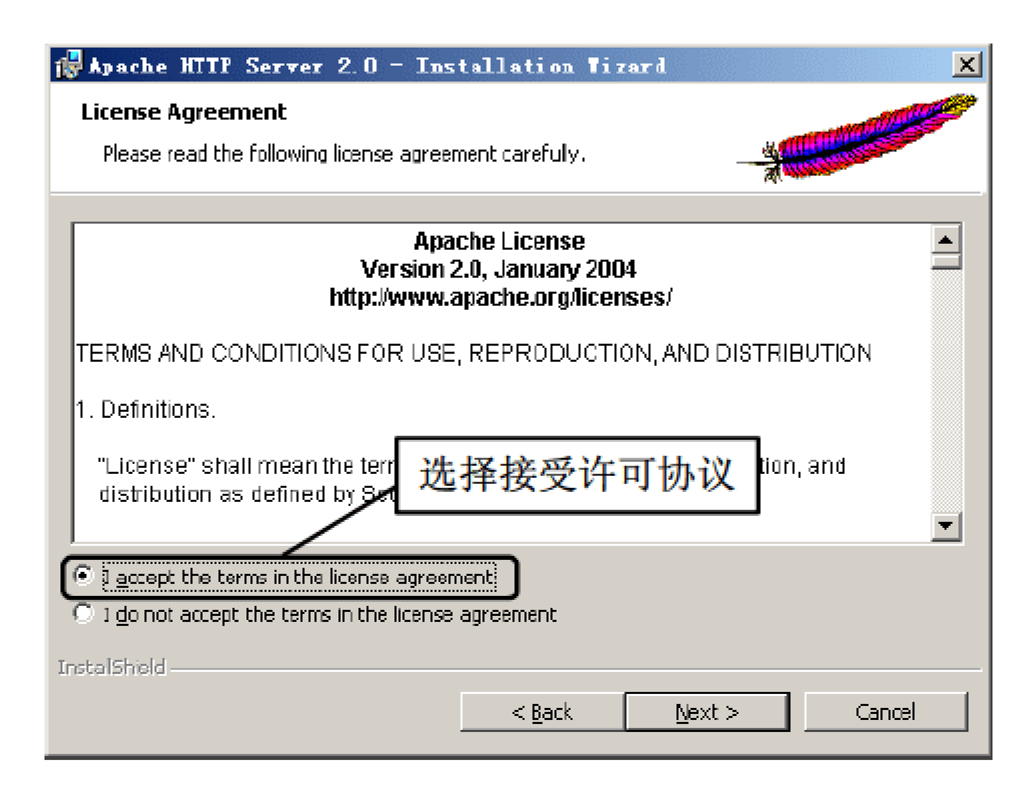


图 1.2　许可协议窗口

（3）打开如图 1.3 所示的 HTTP 服务窗口，单击 Next 按钮，打开如图 1.4 所示的输入服务器信息窗口。输入服务器的相关信息，如网络域名、服务器名和管理员邮箱等，这里可以根据用户的实际情况进行输入。在下方的单选按钮组中，如果选择第一项则可以对任何用户开放 Apache 服务，同时设置服务器的侦听端口为“80”；如果选择第二项则只有本地用户可以连接和使用 Apache 服务。

注意：

如果选择的是第一项，且同时安装了 IIS，那就必须修改 IIS 的默认端口，否则将导致 Apache 无法正常工作。更改 IIS 的默认侦听端口 80，可以在 IIS 的管理器中进行设置，或者停止 IIS 的服务也可以。



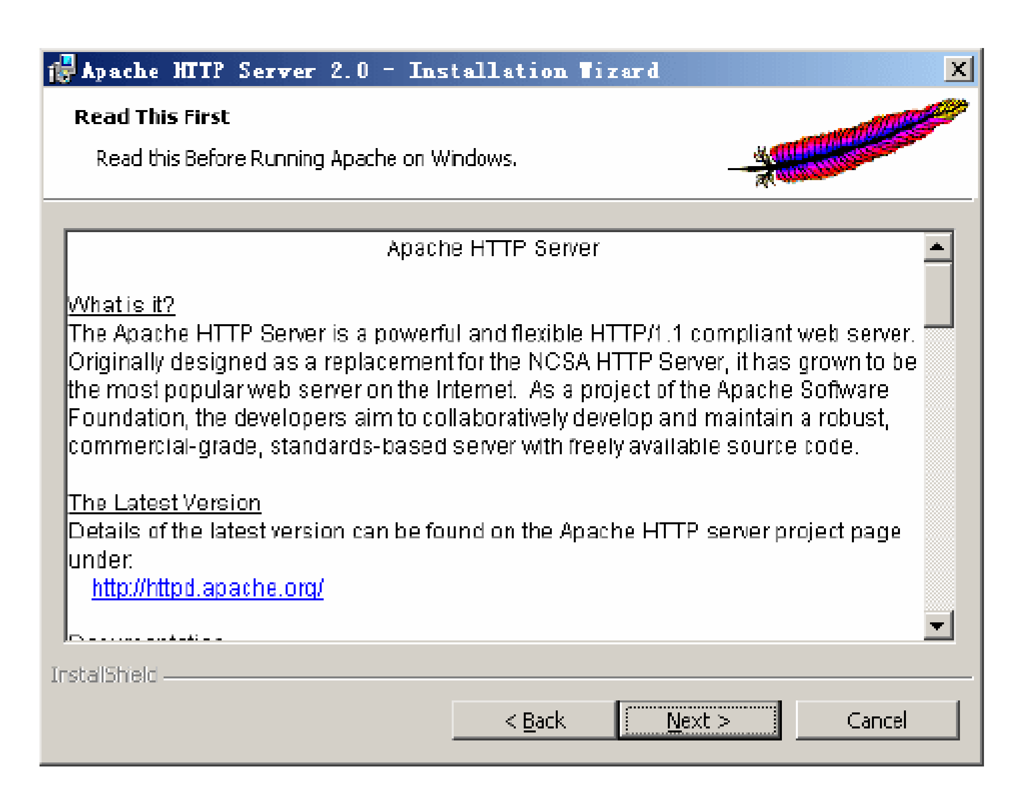


图 1.3　HTTP 服务窗口

Apache HTTP Server 2.0 - Installation Wizard
Server Information
Please enter your server's information.
Network Domain (e.g. somenet.com)
mrsoft
Server Name (e.g. www.somenet.com):
zts
Administrator's Email Address (e.g. webmaster@somenet.com):
mingrisoft@mingrisoft.com
Install Apache HTTP Server 2.0 programs and shortcuts for:
for All Users, on Port 80, as a Service -- Recommended.
only for the Current User, on Port 8080, when started Manually.
InstallShield
< Back　Next >　Cancel

图 1.4　输入服务器信息窗口

（4）设置完成后，单击图 1.4 中的 Next 按钮，打开如图 1.5 所示的选择安装方式窗口。在该窗口中有“Typical”（典型安装）和“Custom”（自定义安装）两种安装方式。通常，用户都选择典型安装方式，单击 Next 按钮。

（5）打开如图 1.6 所示的设置安装的路径窗口，在其中可以选择安装的路径，单击 Change 按钮。

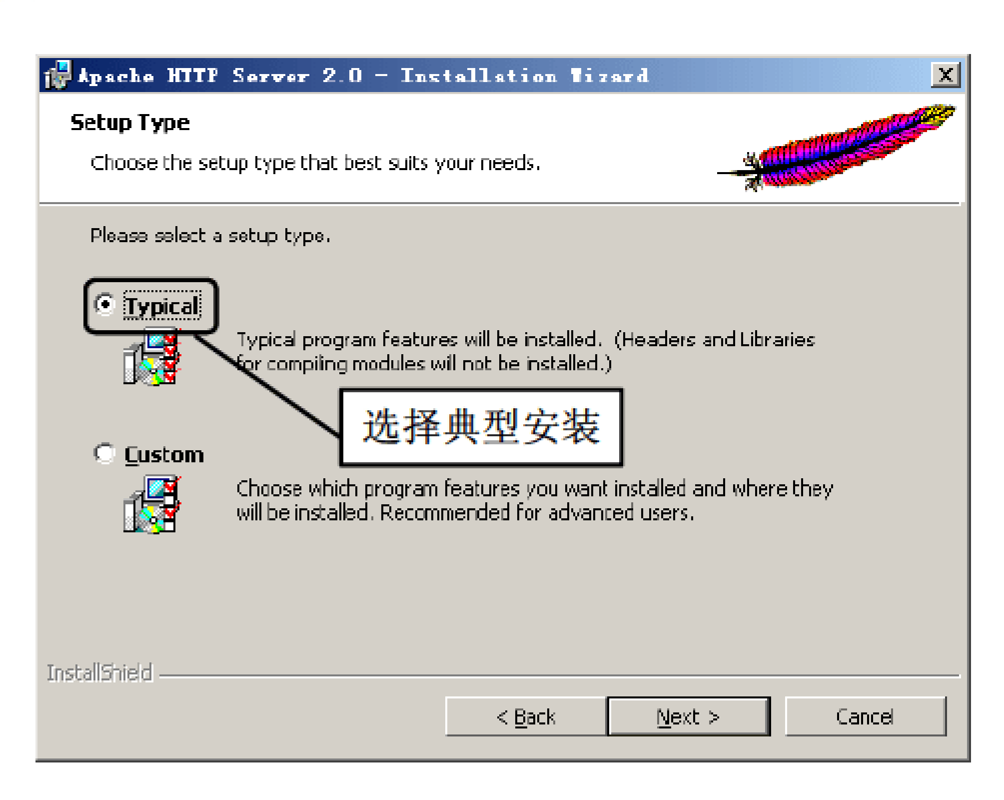


图 1.5　选择安装方式

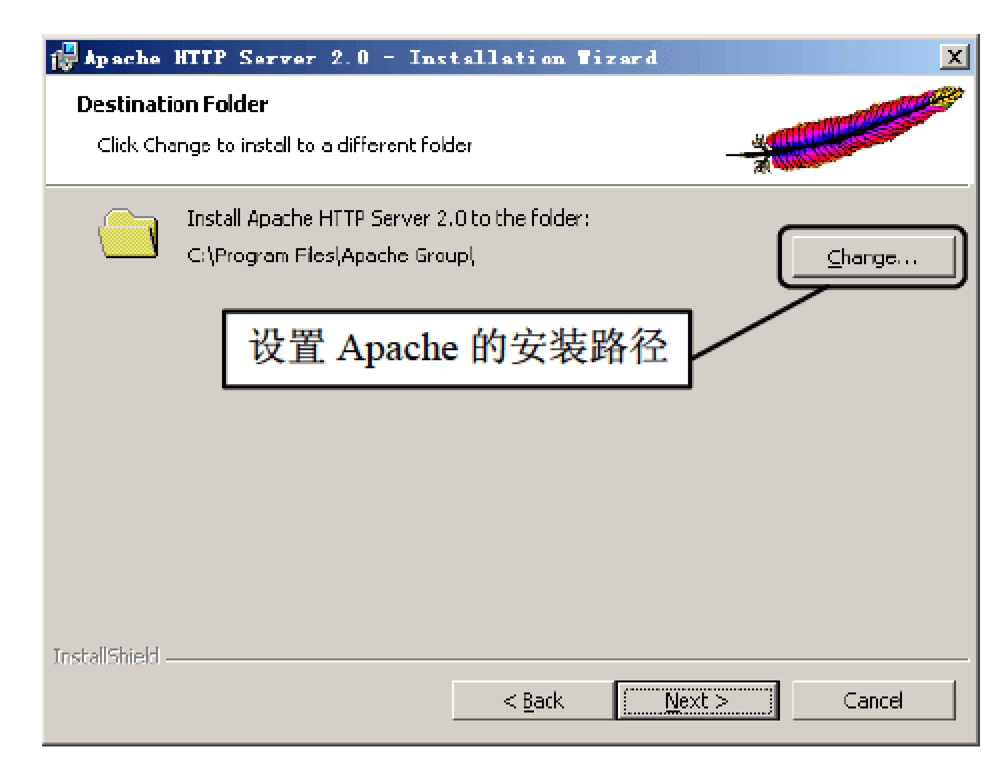


图 1.6　设置安装的路径

（6）打开如图 1.7 所示的修改文件的安装路径窗口，在其中对安装的路径进行修改，然后单击 OK 按钮。

（7）打开如图 1.8 所示的准备安装窗口，单击其中的 Install 按钮。

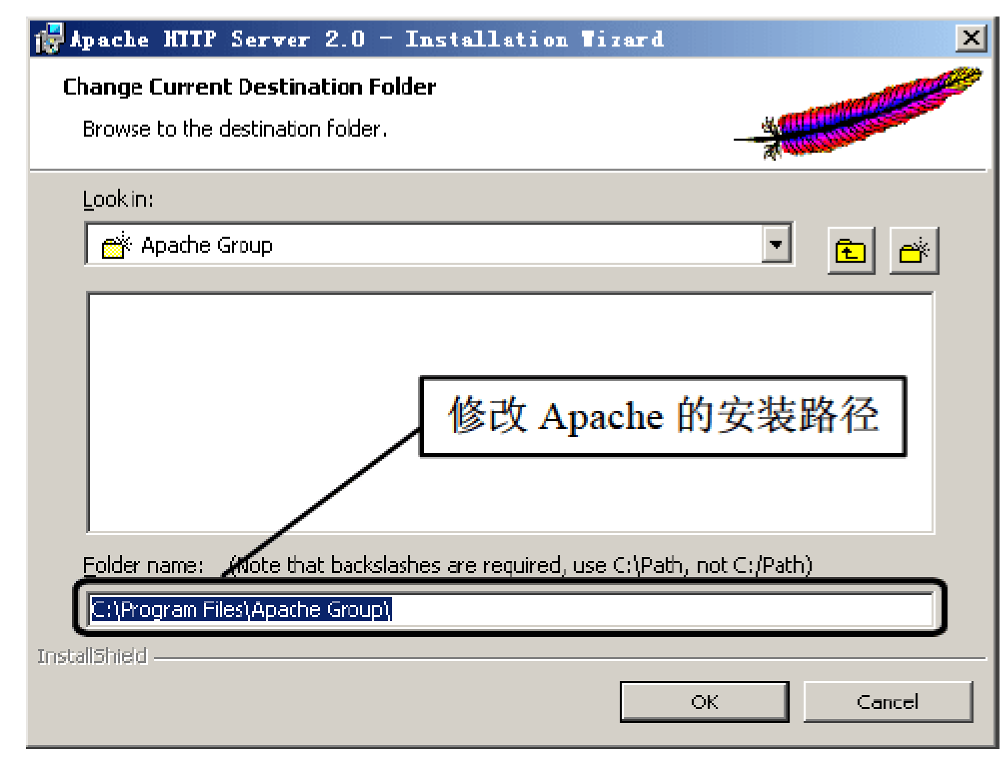


图 1.7　修改安装的路径

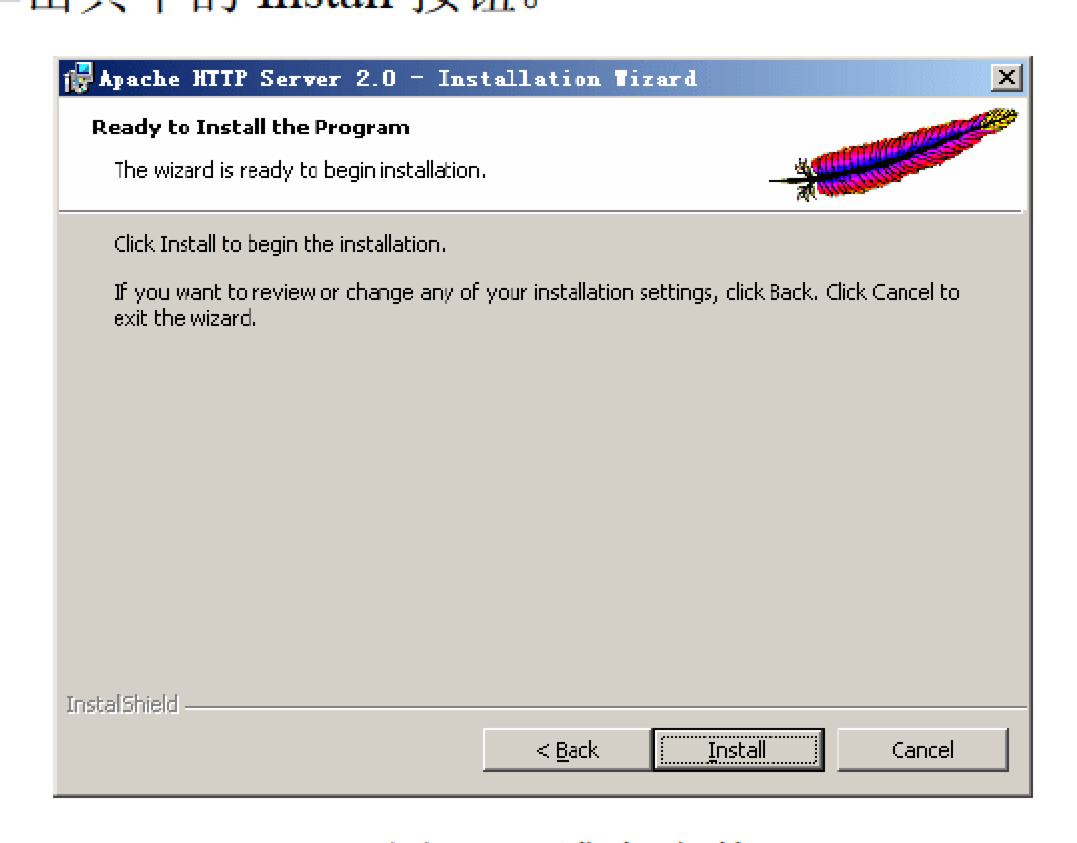


图 1.8　准备安装

Note

（8）打开如图 1.9 所示的开始安装窗口，在其中所有文件复制完成后，Apache 安装完成，打开如图 1.10 所示的窗口，单击 Finish 按钮完成 Apache 的安装。

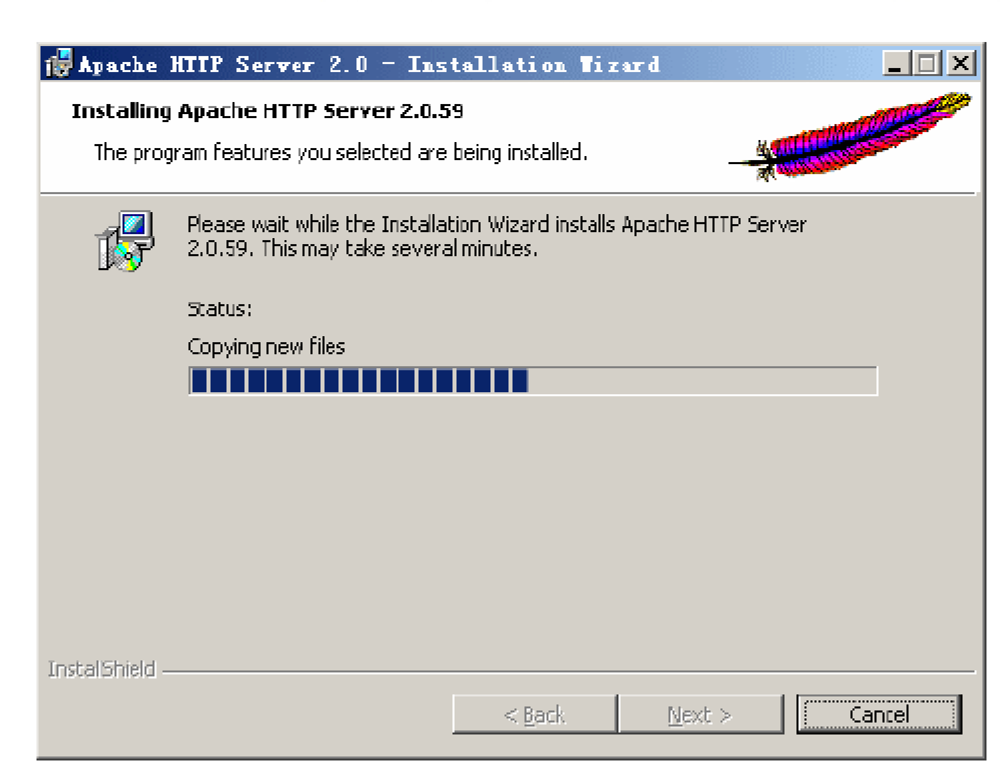



图 1.9　复制文件

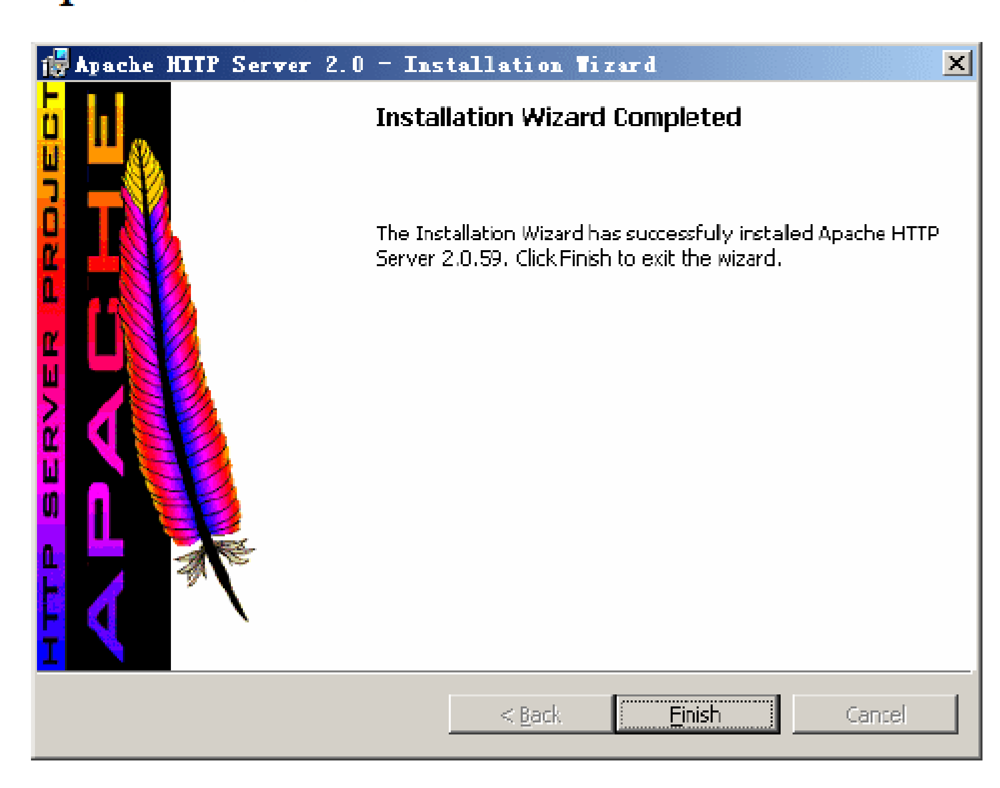


图 1.10　安装完成

（9）接下来测试 Apache 服务是否安装成功。选择“开始”→“所有程序”命令，在弹出的菜单中能够看到 Apache 服务器相关操作列表，同时如果在系统托盘中有一个图标，则表示 Apache 服务已经启动。

（10）单击图标，打开如图 1.11 所示的窗口。在该窗口中，选择 Stop 选项表示停止 Apache 服务器；选择 Start 选项表示启动 Apache 服务器（当 Apache 服务器的当前状态为停止时，该功能可用）；选择 Restart 选项表示重新启动 Apache 服务器。双击图标后，打开如图 1.12 所示的窗口，可以进行启动和停止服务器的各种操作。



图 1.11　左键菜单

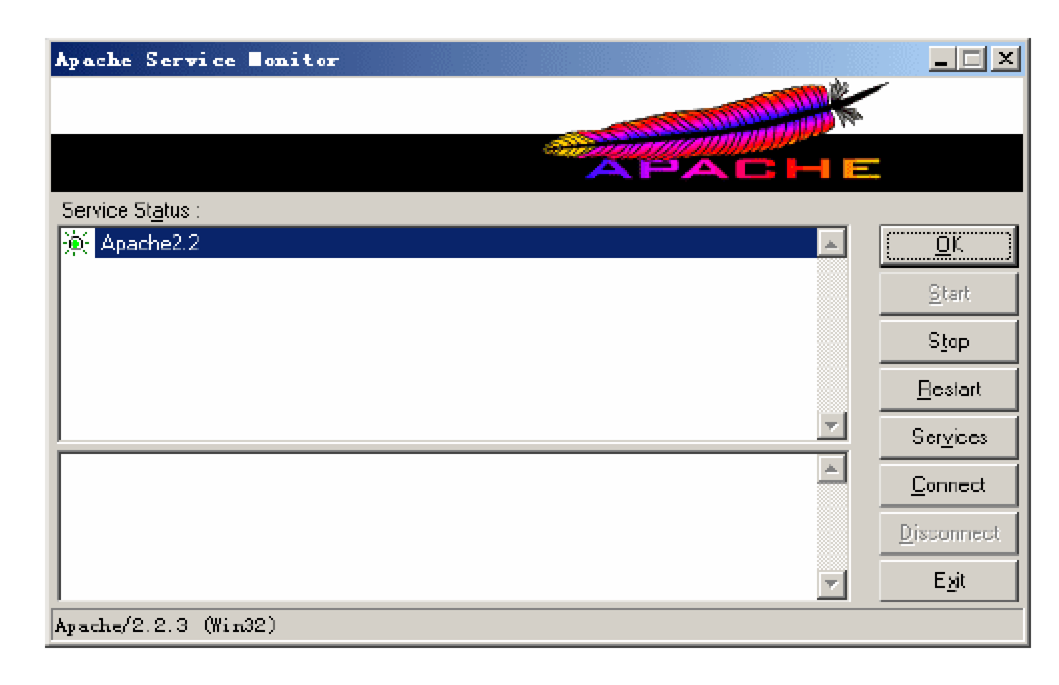
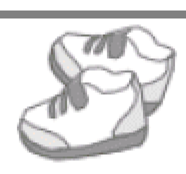


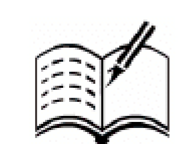
图 1.12　Apache 服务器的监视器

（11）在浏览器中输入“http://localhost/”或“http://127.0.0.1/”，如果能够浏览到如图 1.13 所示的窗口，则说明 Apache 服务器安装成功。



图 1.13　测试 Apache 服务器是否安装成功





Apache 服务器安装成功后，接下来对 Apache 服务器进行配置，以便 Apache 服务器能够识别 PHP 文件。

具体步骤如下：

（1）配置 Apache 服务器主要是在 Apache 安装目录下的 conf 子目录中的 httpd.conf 文件中进行的，找到该文件并用记事本等文本编辑器打开。

（2）定位到“DocumentRoot”一行，可以将该路径修改为合适的路径。例如，设置为“DocumentRoot"C:\Apache\Apache2\webpage"”表示 Apache 服务器的虚拟目录为“C:\Apache\Apache2\webpage”。

（3）将光标定位到下面这行代码：DirectoryIndex index.html index.html.var，在其后面添加一个 PHP 默认页，通常是 index.php，表示当访问该服务器时如果未指定要访问的 PHP 文件，则默认访问 index.php 文件，更改后的代码如下。

```
DirectoryIndex index.html index.html.var index.php
```

（4）为了使 Apache 能够识别 PHP 的扩展名，搜索并定位到 httpd.conf 文件的下面这个部分。

```
<Directory "C:/Apache/Apache2/cgi-bin">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>
```

在后面添加如下两行代码。

```
AddType application/x-httpd-php .php .phtml .php3 .php4
AddType application/x-httpd-php-source .phps
```

（5）为了能够使用模块功能，模块通常以 DSO 方式构建，需要定位到如下这行代码：

```
#LoadModule ssl_module modules/mod_ssl.so
```

使得能够在使用前获得指令的功能，然后以 module 方式加载 PHP，指向 PHP 5.0 目录下的 php5apache2.dll 文件，添加如下代码。

```
LoadModule php5_module c:\php5\php5apache2.dll
```

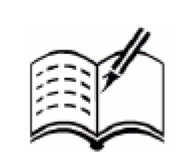
2. Windows 下 MySQL 的安装配置

PHP 能够支持市面上的大多数数据库，例如 MySQL、Access、SQLServer、Oracle 等。MySQL 数据库与 PHP 是程序开发人员公认的黄金搭档，这不仅因为 MySQL 是完全网络化的跨平台关系型数据库系统，还因为 MySQL 是具有客户机/服务器体系结构的分布式数据库管理系统，该数据库具有功能性强、使用简捷、管理方便、运行速度快、版本升级快、安全性高等优点，而且是完全免费的，从其官方网站 http://www.MySQL.com 即可免费下载最新版本的安装包。

在 Windows 下实现 MySQL 的安装配置的操作步骤如下。

（1）如果下载的是 Windows Essentials（x86），例如 MySQL-4.1.11-essential-win.exe，则直接双击该文件安装即可。安装完毕后，选择“开始”→“运行”命令，在弹出对话框的文本框中输入“cmd”，进入“命令提示符”界面，输入“MySQLd-nt.exe”，即可启动 MySQL 服务。如果还想详细地配置 MySQL，则需要进入“C:\MySQL\bin”目录，运行 MySQLInstanceConfig.exe，然后按提示操作即可。





Note

（2）如果下载的是 Windows (x86)，例如 MySQL-essential-5.0.24-win32.msi，则解压该文件后双击 setup.exe 安装程序即可逐步完成 MySQL 的安装。

（3）如果下载的是 Without noinstall (unzip in C:\)，则直接将该文件解压到指定的安装目录下。选择“开始”→“运行”命令，在弹出对话框的文本框中输入“cmd”，进入“命令提示符”界面，输入“MySQLd-nt.exe”命令，即可启动 MySQL 服务器。

3. Windows 下 PHP 的安装配置

架设基于 PHP 的 Web 服务器，安装 PHP 是必须的。由于 PHP 的代码是公开的，所以其升级速度较快。安装 PHP 之前应从官方网站 http://www.php.net/下载最新版本的 PHP 安装程序。下面以 php-5.1.4-Win32.zip 为例介绍 PHP 的安装配置方法。

在 Windows 下实现 PHP 的安装配置的操作步骤如下。

（1）成功下载 PHP 的安装包后，首先应对其进行解压，如解压到 C:\php5\目录下，如图 1.14 所示。

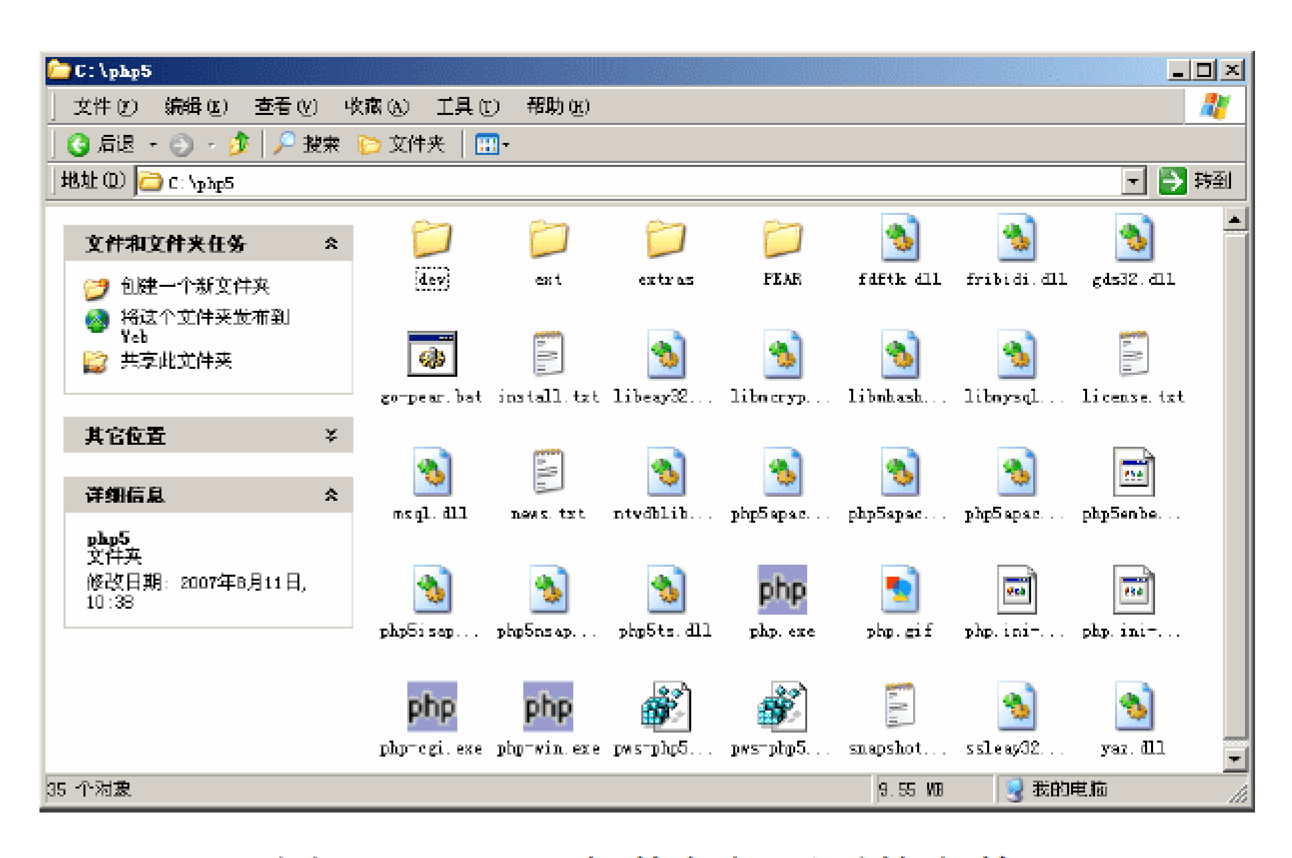


图 1.14　PHP 包装包解压后的文件

（2）将 C:\php5\目录下的所有.dll 文件复制到 C:\Windows\system32\（如果是 Windows 2000 操作系统，则为 C:\WINNT\system32\）目录下。

（3）将 C:\php5\目录下的 php.ini-dist 文件复制到 C:\Windows\（如果是 Windows 2000 操作系统，则为 C:\WINNT\）目录下，然后将 php.ini-dist 重命名为 php.ini，并用记事本打开该文件进行编辑，具体编辑方式如下。

首先，定位到该文件 extension_dir=".\"所在行，将路径改为"C:\php5\ext"。代码如下。

```
extension_dir="C:\php5\ext"
```

然后，查找定位 Windows Extensions，将下面两行代码前面的注释（即分号标识符）去掉。

```
; extension=php_gd2.dll
; extension=php_MySQL.dll
```

这样 PHP 就可以支持 GD2 函数库和 MySQL 相关函数了，如果想让 PHP 支持其他功能函数，同样需要将与这些操作函数相关的.dll 文件前的注释去掉。

注意：

以上为 PHP 的常规配置，在 PHP 的官方论坛上可以查看到更多的详细配置。由于 PHP 的版本有多种，所以 php.ini 的配置也存在差别，具体使用时应根据实际情况而定。





（4）当 PHP 配置完成后需要重新启动 Apache 服务器，然后编写 test.php 文件并输入如下代码。

```
<?php
  phpinfo();
?>
```

将文件保存到 Apache 安装目录下的 webpage 子目录下，然后在浏览器中输入“http://127.0.0.1/test.php”，如果显示如图 1.15 所示的页面则说明 PHP 配置成功。

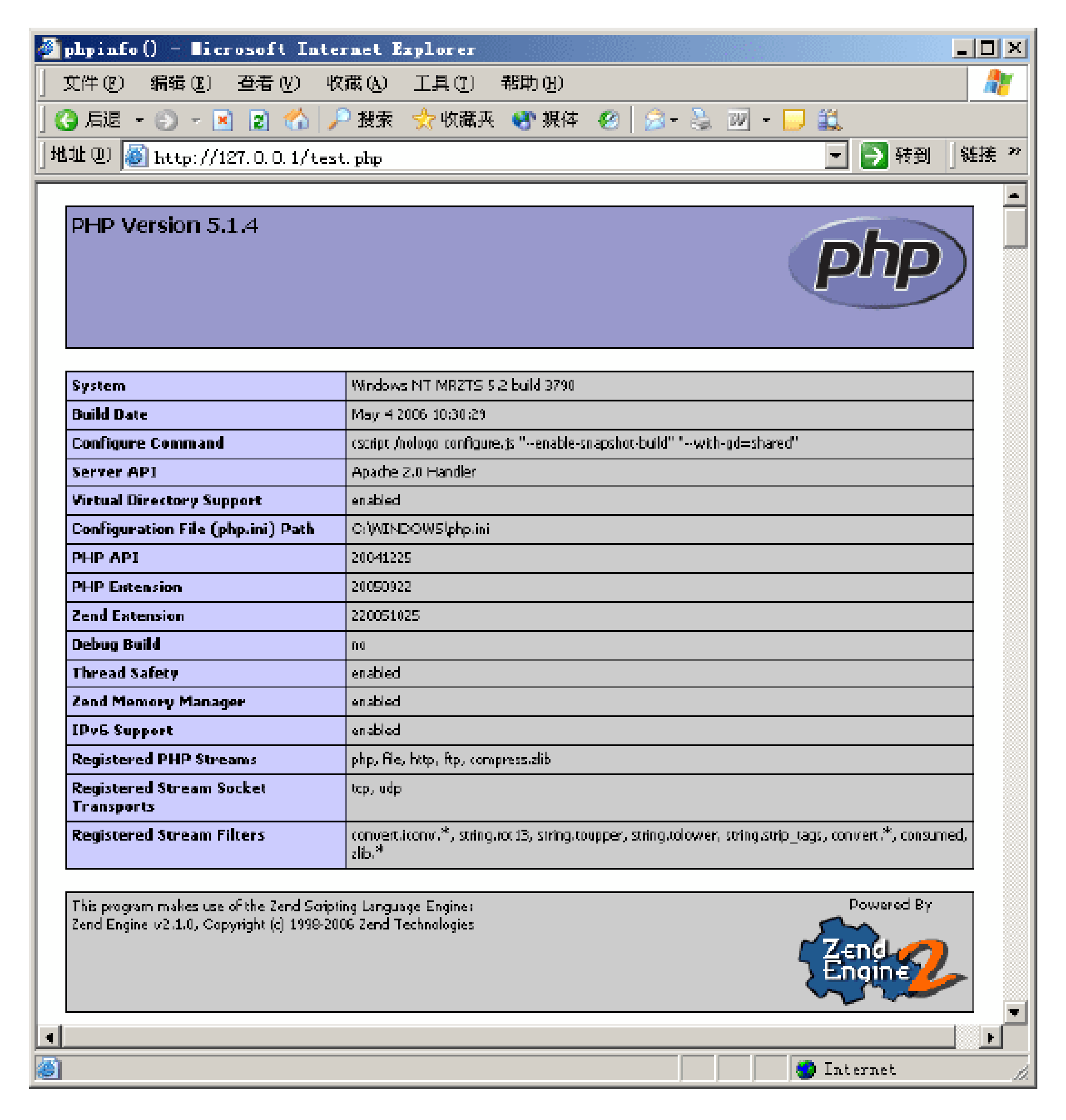


图 1.15　配置成功

技巧：

安装文件的路径也要遵循一定的客观原则，为了避免在 Windows 和 Linux 间移植程序时带来的不便，当选择 D:\usr\local\php 的目录时要和在 Linux 下的安装目录相匹配。建议最好不要选择中间有空格的目录，如 E:\program Files\PHP，这样做会导致一些未知错误或崩溃情况发生。

问题 2　如何在 Linux 下配置 PHP 开发环境？

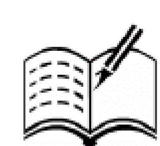
问题阐述

在 Linux 下搭建 PHP 开发环境比 Windows 下要复杂很多。除了需要安装 Apache、PHP 等软件外，还要安装一些相关工具，设置必要参数。另外，如果要使用 PHP 扩展库，还要进行编译。在 Linux 下配置 PHP 开发环境的具体的过程是什么？

专家解答

首先打开 Linux 终端（Linux 下几乎所有的软件都需要在终端下安装）。在 RedHat 9 中选择“主菜单”→“系统工具”→“终端”命令。







1. Linux 下安装 Apache

首先到官方网站 http://www.apache.org 下载 Linux 下“httpd2.2.8.tar.gz”的压缩包。

在 Linux 下安装和配置 Apache 服务器的操作步骤如下。

（1）进入 Apache 安装文件的目录下，如/usr/local/work。

```
cd /usr/local/work/
```

（2）解压安装包，完成后进入 httpd2.2.8 目录中。

```
tar xfz httpd2.2.8.tar.gz
cd httd2.2.8
```

（3）建立 makefile，将 Apache 服务器安装到 user/local/Apache2 目录下。

```
./configure –prefix=/usr/local/Apache2 –enable-module=so
```

（4）编译文件。

```
make
```

（5）开始安装。

```
make install
```

（6）安装完成后，在终端窗口中输入如下命令启动或重启 Apache 服务。

```
/usr/local/Apache2/bin/Apachectl start
/usr/local/Apache2/bin/Apachectl restart
```

（7）打开 Mozilla 浏览器，在地址栏中输入“http://localhost/”，回车后如果看到如图 1.16 所示的页面，则说明 Apache 服务器安装成功。

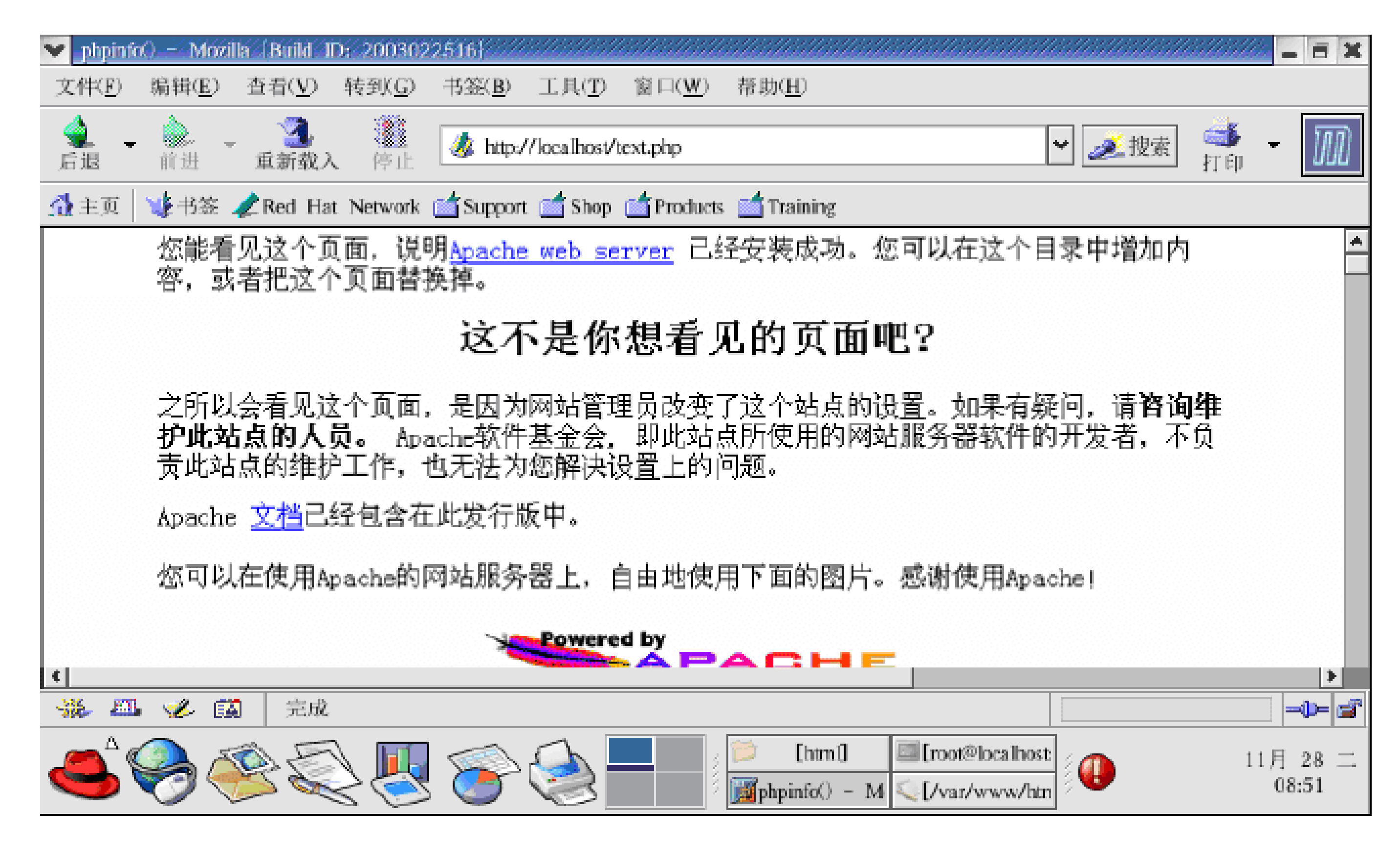


图 1.16　Linux 下的 Apache 服务器安装

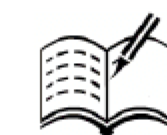
2. Linux 下安装 PHP

在安装 PHP5 之前，需要首先查看 libxml 的版本号。如果 libxml 版本号小于 2.5.10，则需要先安装 libxml 的高版本。安装 libxml 和 PHP5 的步骤如下（如果不需要安装 libxml，则直接跳到 PHP5 的安装步骤即可）。

（1）将 libxml 和 PHP5 复制到/usr/local/work/目录下，并进入该目录中。

```
cd /usr/local/work
```





（2）分别将 libxml2 和 PHP 解压。

```
tar xfz libxml2-2.6.62.tar.gz
tar xfz PHP-5.2.5.tar.gz
```

（3）进入 libxml 目录，建立 makfile，将 libxml 安装到/usr/local/libxml2 目录下。

```
cd libxml2-2.6.62
./configure –prefix=/usr/local/libxml2
```

（4）编译文件。

```
make
```

（5）开始安装。

```
make install
```

（6）libxml2 安装完毕，开始安装 PHP5，进入 php-5.2.5 目录下。

```
cd php-5.2.5
```

（7）建立 makefile。

```
./configure –with-apxs2=/usr/local/Apache2/bin/apxs \
--with-mysql=/usr/local/mysql \
--with-libxml-dir=/usr/local/libxml2 \
--enable-sockets \
--enable-mbstring \
--enable-soap
```

（8）开始编译。

```
make
```

（9）开始安装。

```
make install
```

（10）复制 php.ini-dist 或 php.ini-recommended 到/usr/local/lib 目录，并命名为 php.ini。

```
cp php.ini-dist /usr/local/lib/php.ini
```

（11）更改 httpd.conf 文件相关设置，该文件位于/usr/local/Apache2/conf 目录中。找到该文件中的如下指令行：

```
AddType application/x-gzip .gz .tgz
```

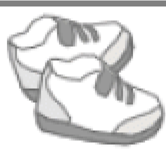
在该指令后加入如下指令：

```
AddType application/x-httpd-php .php
```

（12）重新启动 Apache，并在 Apache 主目录下建立文件 phpinfo.php。

```
<?php
phpinfo();
?>
```

（13）在 Mozilla 浏览器中输入“http://localhost/phpinfo.php”，如果出现如图 1.17 所示的页面，则表明 PHP 安装成功。



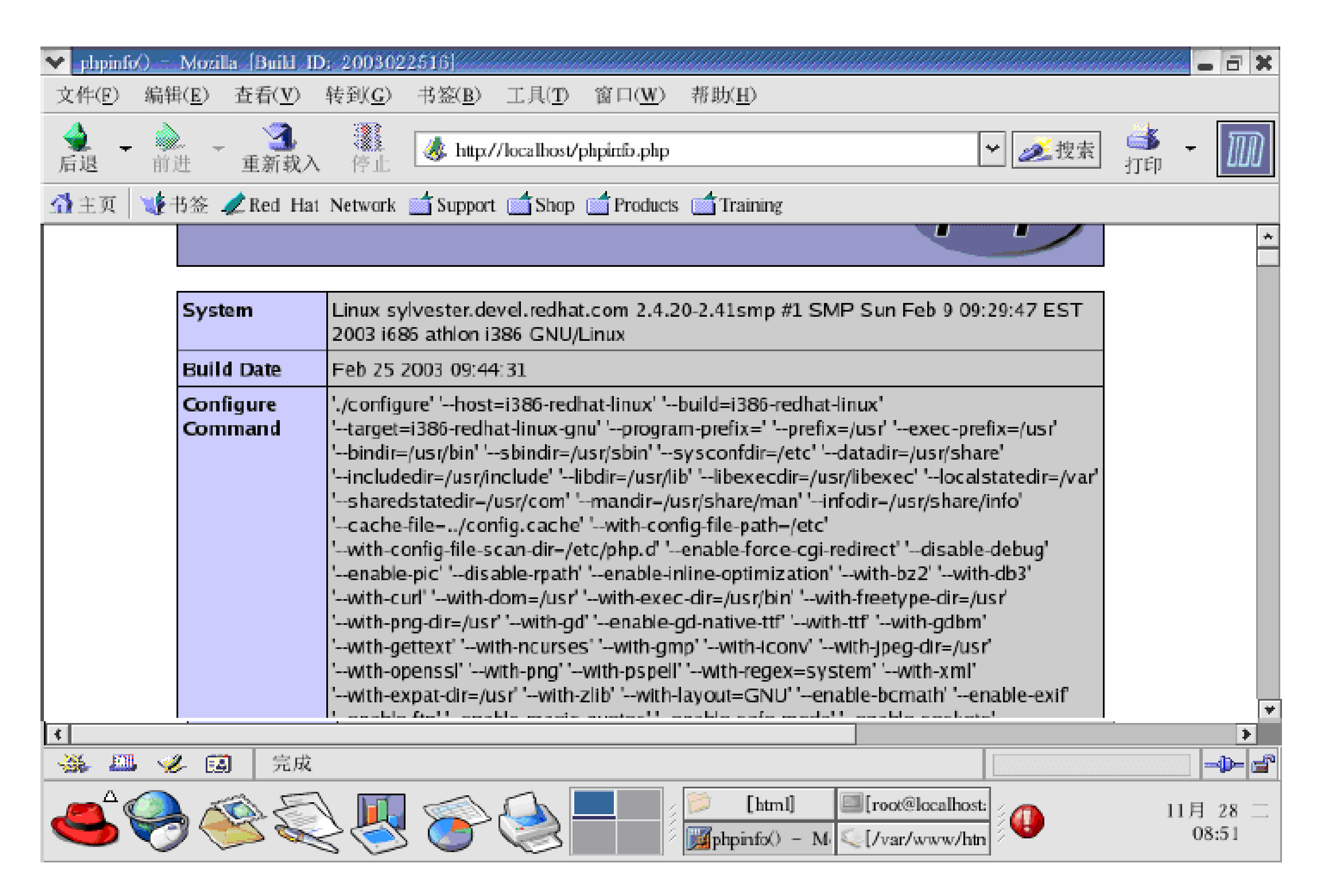


图 1.17　phpinfo 信息

3. Linux 下安装 MySQL

在 Linux 下安装 MySQL，需要到官方网站 http://www.MySQL.com 下载 Linux 下 MySQL 的安装包。Linux 下 MySQL 的安装配置步骤如下：

（1）将下载的 mysql-5.1.56.tar.gz 复制到/softs（此为自定义文件夹，用于存储压缩包）下。

（2）添加用户和用户群组。

```
groupadd mysql
useradd -g mysql mysql
```

（3）进入存放安装文件的文件夹 softs（自行创建的一个存放安装文件的文件夹）。

```
cd /softs
```

（4）解压数据库文件。

```
tar -vzxf mysql-5.1.56.tar.gz
```

（5）进入解压后的 mysql 文件夹。

```
cd mysql-5.1.56
```

（6）安装配置文件。

```
./configure --prefix=/usr/local/mysql
```

（7）编译文件。

```
make
```

（8）安装编译文件。

```
make install
```

（9）进入 mysql 安装目录。

```
cd /usr/local/mysql
```

（10）提供新的文件所有者。

```
chown -R mysql .
```



（11）提供新的文件群组所有者。

```
chgrp -R mysql .
```

（12）初始化数据目录。

```
cd /usr/local/mysql/bin
./mysql_install_db --user=mysql
```

（13）启动 mysql。

```
./mysqld_safe &
```

（14）设置 mysql 数据库密码。

```
./mysqladmin -u root password 111
```

（15）登录 mysql 数据库。

```
./mysql -u root -p
```

（16）关闭数据库（这个很特殊，要注意进程问题）。

```
cd /usr/local/mysql/bin
./mysqladmin -u root shutdown -p
```

（17）创建新的配置文件。

```
cd /usr/local/mysql/share/mysql
cp my-huge.cnf /etc/my.cnf
chmod 777 /etc/my.cnf
```

注意：

在创建新的配置文件时，不同的内存需要使用不同的配置文件来创建新的配置文件。

```
my-small.cnf        //适用于小于 64MB 的服务器
my-medium.cnf       //适用于物理内存在 28MB 到 64MB，或者物理内存在 128MB 以上，但要运行
                     其他程序的服务器
my-large.cnf        //适用于物理内存在 512MB 以上的专用于数据库的机器
my-huge.cnf         //适用于物理内存 1GB 到 2GB 的专用于数据库的机器
my-innodb-heavy-4G.cnf //适用于物理内存在 4GB 及以上专用于数据库的机器，且需要复杂查询
```

（18）修改数据库字符集，在新创建的/etc/my.cnf 中进行添加，注意其添加的位置。

在[client]下添加：

```
default-character-set=utf8
```

在[mysqld]下添加：

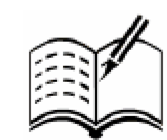
```
default-character-set=utf8
init_connect='SET NAMES utf8'
```

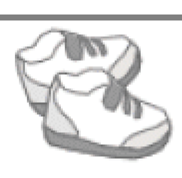
设置完毕，登录数据库，使用下面的命令查询字符集的状态。

```
status
```

（19）对数据库进行备份和还原。

```
mysqldump -u root -p root -R  数据库名>c:/data.sql（备份，c:/为自定义的一个存储位置，data.sql
```





为自定义的存储文件名称）

```
mysql -uroot -proot -D 数据库名<c:/data.sql（还原，c:/data.sql 为备份数据库文件所存储的位置）
```

（20）重置数据库密码，修改 root 用户的密码。

```
mysqladmin -uroot -p 旧密码 password 新密码
```





注意：

Linux 下启动和停止 MySQL 服务器的命令是（使用 rpm 包安装的 mysql，可以使用下面的方式进行 mysql 的启动、停止和重启）：

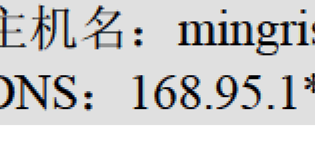
```
service mysqld start
service mysqld stop
service mysqld restart
```

4. Linux 下发布网站

Linux 下发布基于 PHP 的网站，首先需要配置 PHP 的运行环境，其次需要对网卡参数进行设定。本实例将以发布“读者之家”网站为例讲解 Linux 下如何实现网站的发布。

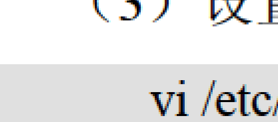
（1）假设已经申请到了以下网络参数：

```
IP：192.168.1.*
Netmask：255.255.255.*
Network：192.168.1.0
Broadcast：192.168.1.*
Gateway：192.168.1.*
主机名：mingrisoft
DNS：168.95.1*
```



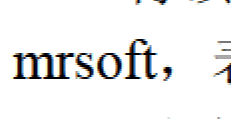
（2）将“读者之家”网站的所有文件复制到 Apache 主目录下。

（3）设置主机名称。在终端窗口中输入如下命令编辑/etc/sysconfig/network 文件。

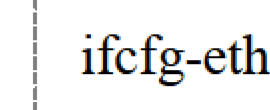


```
vi /etc/sysconfig/network
```

将该文件中的参数 NETWORKING 设置为 yes，表示启动网络，将参数 HOSTNAME 设置 mrsoft，表示设置主机名为“mrsoft”。



（4）设置网卡参数。在终端窗口中输入如下命令编辑文件/etc/sysconfig/network-scripts/ifcfg-eth0。



```
vi  /etc/sysconfig/network-scripts/ifcfg-eth0
```

按如下形式设置该文件的相关参数：



DEVICE=eth0　　设置网卡名称，要与 ifcfg-eth0 对应。



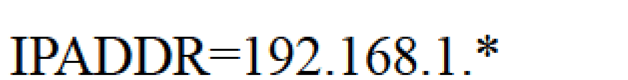
ONBOOT=yes　　指定在开机时启动网卡。



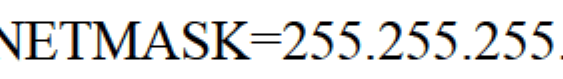
BOOTPROTO=static　　设定启动时获取 IP 的方式。



IPADDR=192.168.1.*　　设定服务器 IP 地址。



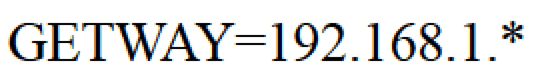
NETMASK=255.255.255.*　　设定子网掩码。



BROADCAST=192.168.1.*　　设定同网段的广播地址。



GETWAY=192.168.1.*　　设定网卡的网关。

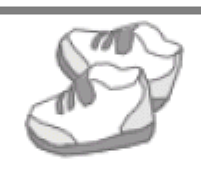


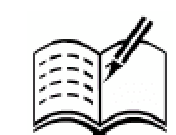
（5）设置 DNS 主机的 IP。在终端编辑/etc/resolv.conf 文件。



```
vi /etc/resolv.conf
```







设置参数 nameserver 的值为 168.95.1.*。

（6）重新启动网络设置。在终端窗口中输入如下命令：

```
/etc/rc.d/inin.d/network restart
ifdow eth0
ifup eth0
```

（7）打开浏览器，在地址栏中输入服务器 IP 地址或域名，如果出现如图 1.18 所示页面，则说明 Linux 下“读者之家”网站发布成功。



图 1.18　在 Linux 下发布网站

专家点评

在 Linux 下配置 PHP 开发环境需要注意以下几个问题：

（1）首先需要打开 Linux 终端（Linux 下几乎所有的软件都需要在终端下安装）。在 RedHat 9 中选择“主菜单”→“系统工具”→“终端”命令。

（2）在安装 PHP5 之前，需要首先查看 libxml 的版本号。如果 libxml 版本号小于 2.5.10，则需要先安装 libxml 高版本。

（3）在 Linux 下配置 Apache 和 PHP，同样需要配置 php.ini 和 httpd.conf。

问题 3　如何搭建 IIS+PHP+MySQL 环境？

问题阐述

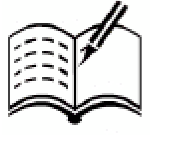
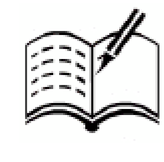
Internet 信息服务器（Internet Information Server，缩写为 IIS）是 Microsoft 的 Web 服务器。它集成于 Windows NT Server 之中，方便易用，为 Web 应用程序提供了功能强大的运行平台。那么该如何应用 IIS 平台来搭建一个 PHP 的开发环境呢？

专家解答

1. 安装 IIS

IIS 目前的最高版本是 Windows Vista 中安装的 IIS 7.0，而 Windows 2003 中的是 IIS 6.0，Windows XP 等使用的则是更低的版本。本章以 Windows 2003 的 IIS 6.0 为例来讲解 IIS 的安装，其他版本的安装和使用基本相同。

（1）选择“开始”→“设置”→“控制面板”命令，在弹出的窗口中选择“添加/删除程

序”窗口。单击左侧的“添加/删除 Windows 组件”按钮，将会弹出“Windows 组件向导”窗口。在该窗口中选中“应用程序服务器”复选框，如图 1.19 所示。

（2）进入“应用程序服务器”窗口，选中“Internet 信息服务(IIS)”复选框，如图 1.20 所示。



图 1.19　选中应用程序服务器

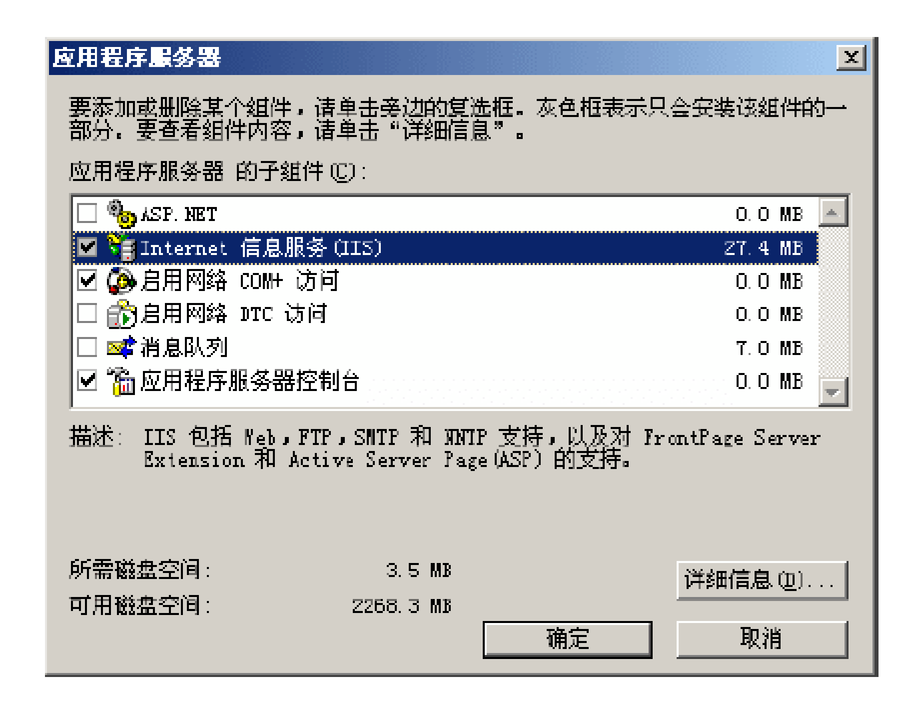


图 1.20　选中 Internet 信息服务器(IIS)

（3）进入“Internet 信息服务(IIS)”窗口，选中“万维网服务”复选框如图 1.21 所示。

（4）进入“万维网服务”窗口，选中“万维网服务”复选框，如图 1.22 所示，然后单击“确定”按钮，开始组件安装。需要注意的是，在安装过程中需要将系统盘放到光驱中。

（5）安装完毕后，就可以使用 IIS 了。如果之前没有安装 Apache 服务器或 Apache 服务器的端口已经被修改了，那么只要打开 IE 浏览器，并在地址栏中输入“http://localhost”就可以看到 IIS 运行页面。

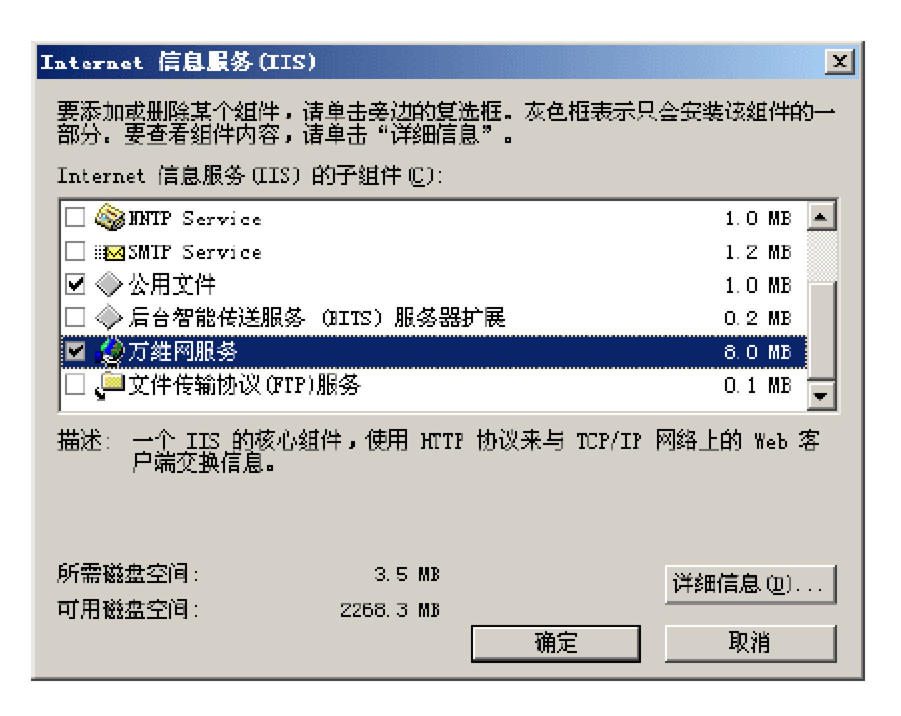


图 1.21　选中万维网服务

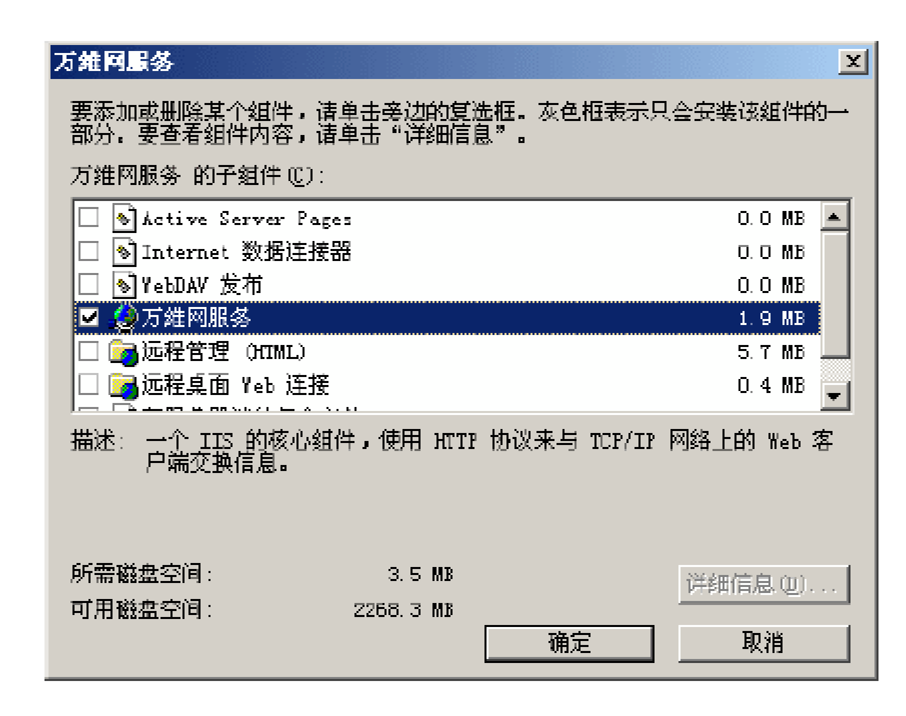


图 1.22　选中万维网服务

2. 配置 IIS

下面对 IIS 进行配置，具体操作如下：

（1）选择“开始”→“设置”→“控制面板”命令，在弹出的窗口中双击“管理工具”文件夹图标。在弹出的窗口中双击刚刚添加的“Internet 信息服务(IIS)管理器”图标，将弹出 IIS 管理器窗口，如图 1.23 所示。

（2）图 1.23 中的默认网站就是 IIS 服务器的文件目录了。右击“默认网站”节点，在弹出的快捷菜单中选择“属性”命令，对网站的属性进行设置，如图 1.24 所示。

（3）最先看到的选项卡是“网站”选项卡，这里主要设置 TCP 端口号，将 TCP 端口号设置为“8080”。设置完成后单击“主目录”标签，进入如图 1.25 所示的“主目录”选项卡。

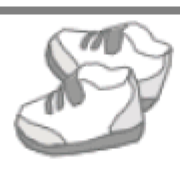




图 1.23　IIS 管理器窗口



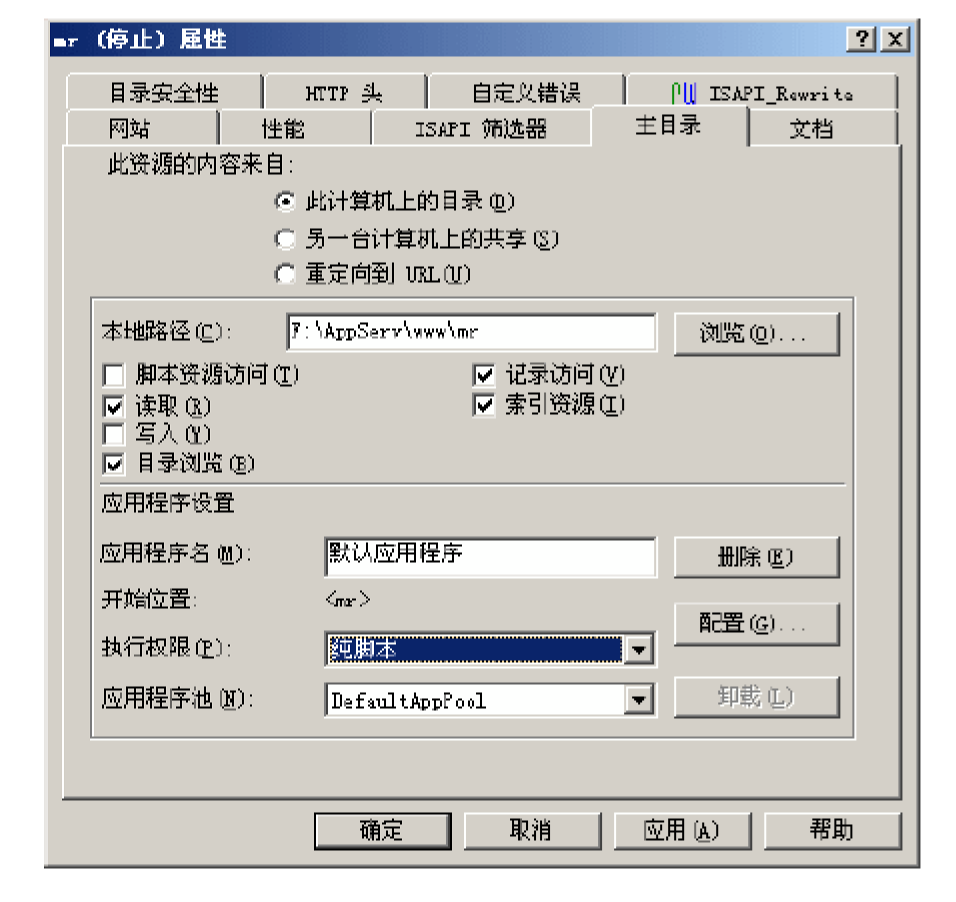




图 1.24　网站属性　　　　图 1.25　主目录选项卡

（4）“主目录”选项卡中需要设置的地方主要有 3 处。第 1 处是本地路径，用户可以自行定义。如 F:\AppServ\www\mr；第 2 处是设置执行权限，设置为纯脚本；第 3 处是添加 PHP 的支持。单击该选项卡中的“配置”按钮，将弹出“应用程序配置”窗口，如图 1.26 所示。

（5）可以看到，在映射选项卡中可以添加、编辑应用程序扩展。单击“添加”按钮，将添加如图 1.27 所示的信息。设置完毕后，单击“确定”按钮返回。

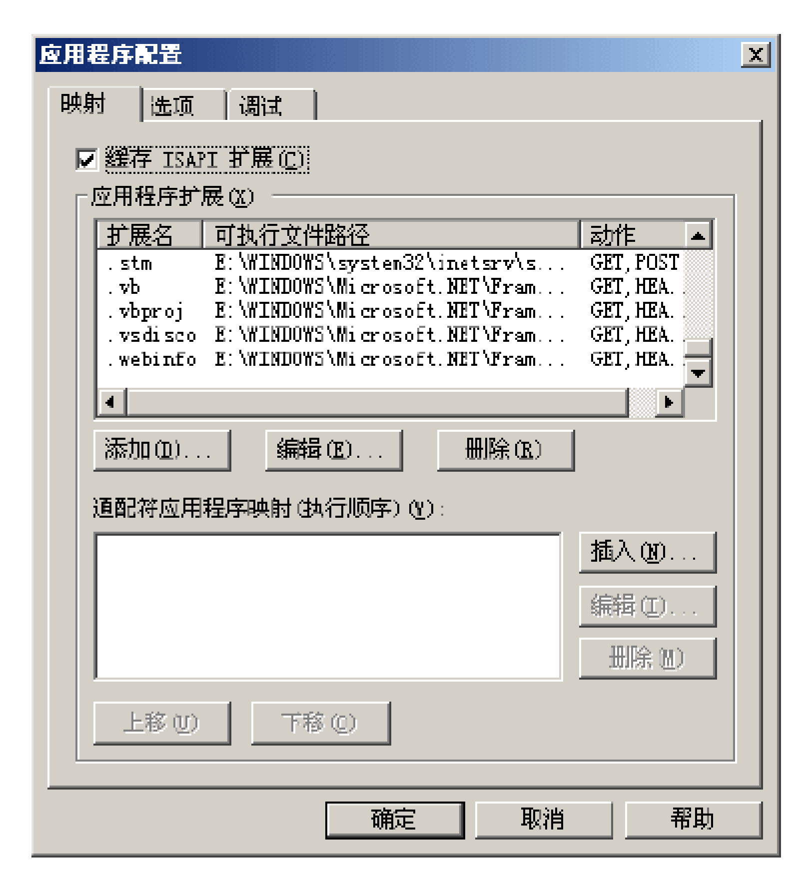


图 1.26　应用程序配置



图 1.27　添加 PHP 扩展

（6）打开“文档”选项卡，添加一个.php 的默认页，如图 1.28 所示。

（7）关闭“默认网站 属性”面板，在 IIS 管理器中选择“Web 服务扩展”选项。在右侧窗口中选择“所有未知 ISAPI 扩展”服务，单击左侧的“允许”按钮，启动该服务。设置后的窗口如图 1.29 所示。






（8）关闭所有窗口，重启 IIS 服务器。

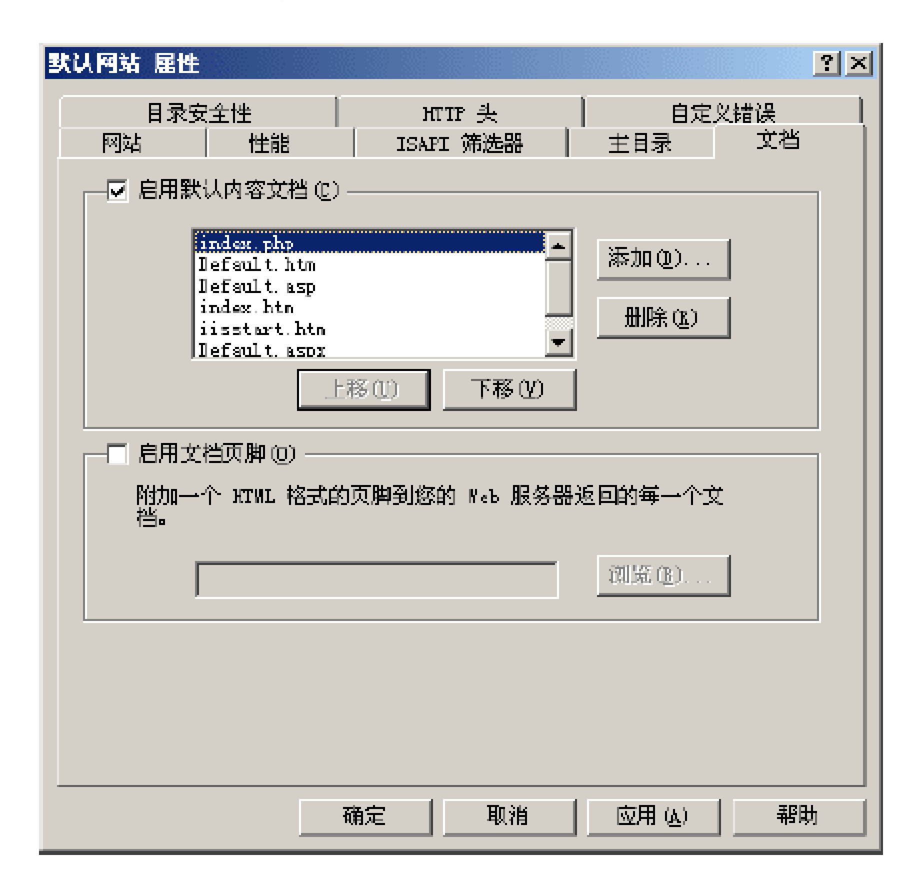


图 1.28　添加默认页



图 1.29　开启 ISAPI 扩展

3. 配置 PHP

（1）将 PHP5 的安装文件 php-5.1.1-Win32.zip 解压到相应目录（如 C:\php\、F:\php\等），这里将其放置在 F:\php\目录下。

（2）将 F:\php\目录下的 libMySQL.dll 文件复制到 C:\Windows\system32\（如果是 Windows 2000，则为 C:\WINNT\system32\）目录下。

（3）将 F:\php\目录下的 php.ini-dist 文件复制到 C:\Windows\（如果是 Windows 2000，则为 C:\WINNT\）目录下。然后将 php.ini-dist 重命名为 php.ini。

（4）打开 php.ini 文件并定位到 extension_dir = "./"这一行，并修改为 extension_dir=" F:\php\ext "。

（5）定位到“;extension=php_MySQL.dll”这一行，将前面的分号“;”去掉。这样，PHP 就可以支持 MySQL 数据库了。如果想要加载其他动态库，其方法与此相同。

（6）保存配置。

创建 phpinfo.php 文件，将其保存在 IIS 主目录中指定的文件夹下，测试 PHP 与 IIS 是否配置成功。代码如下：

```
<?php   phpinfo() ;    ?>
```

（7）在浏览器中输入“http://localhost:8080/phpinfo.php”，如果正确，则显示如图 1.30 所示的页面。

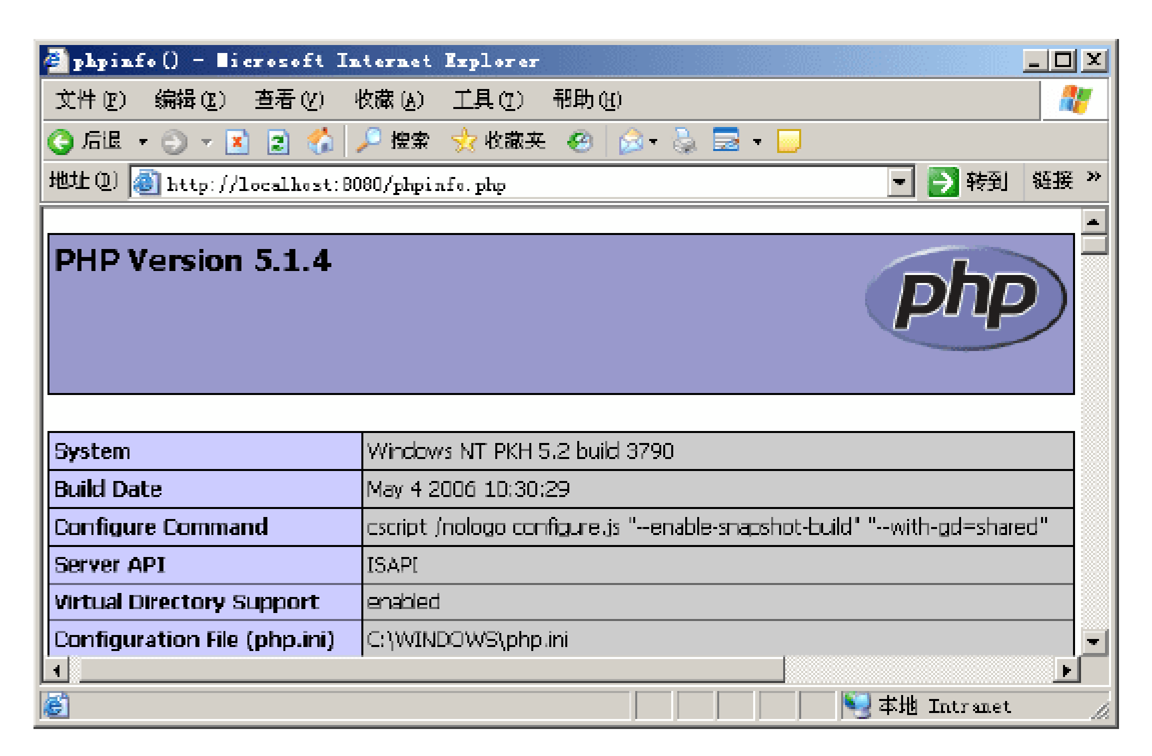
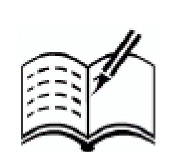


图 1.30　测试成功





4. 安装 MySQL

MySQL 是一款备受用户欢迎的数据库，由于其具有开源的特性，所以市场占有率较高，得到 PHP 开发者的青睐，一直被认为是 PHP 的最佳搭档。MySQL 数据库的安装步骤如下：

（1）双击运行下载后的程序（mysql-5.5.12-win32.msi），如图 1.31 所示。

（2）单击“运行”按钮，弹出如图 1.32 所示的对话框。



图 1.31　询问是否打开文件

图 1.32　开始运行 MySQL 安装向导

（3）单击 Next 按钮，弹出如图 1.33 所示的对话框。

（4）单击 Next 按钮，弹出如图 1.34 所示的对话框。

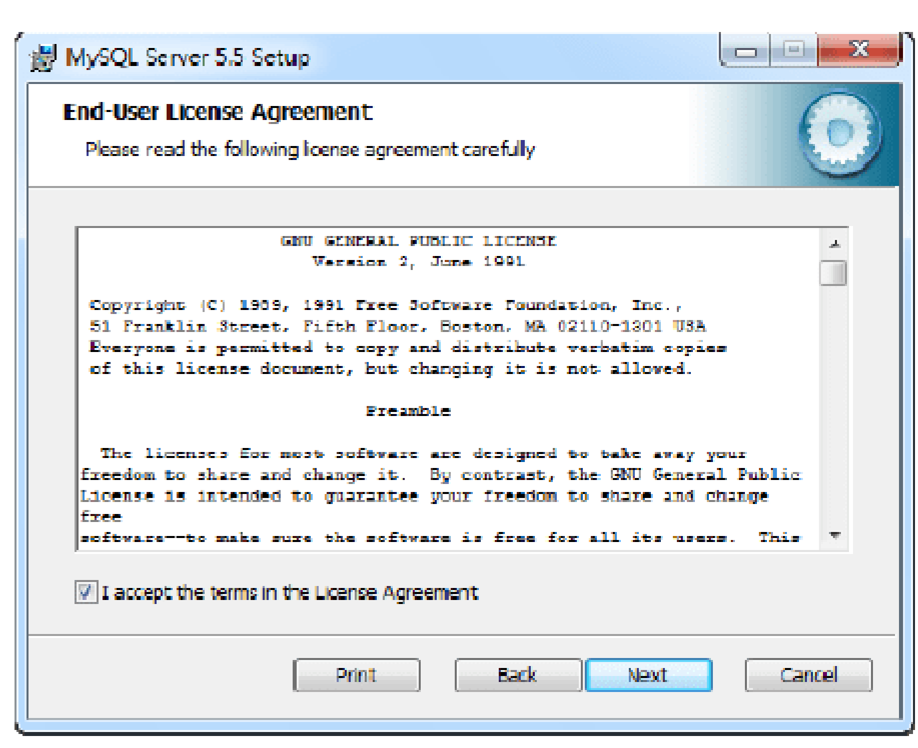


图 1.33　询问是否接受协议

图 1.34　选择安装类型

（5）单击 Typical 按钮，弹出如图 1.35 所示的对话框。

（6）单击 Install 按钮开始安装，安装进度如图 1.36 所示。

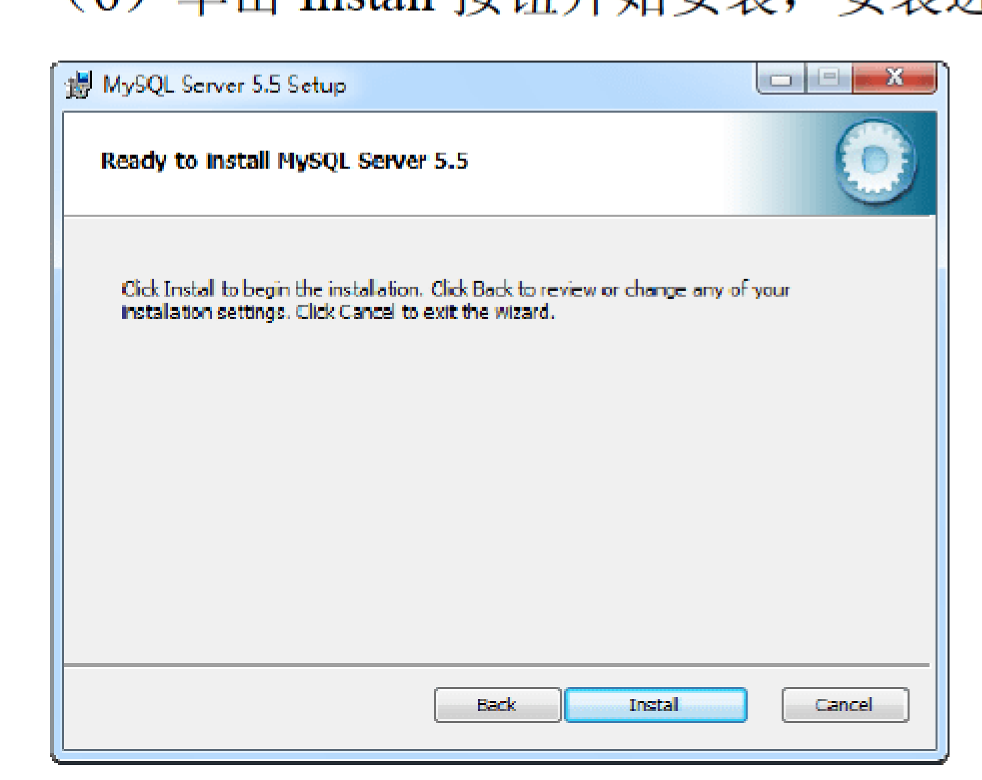


图 1.35　确认前面各选择步骤的对话框

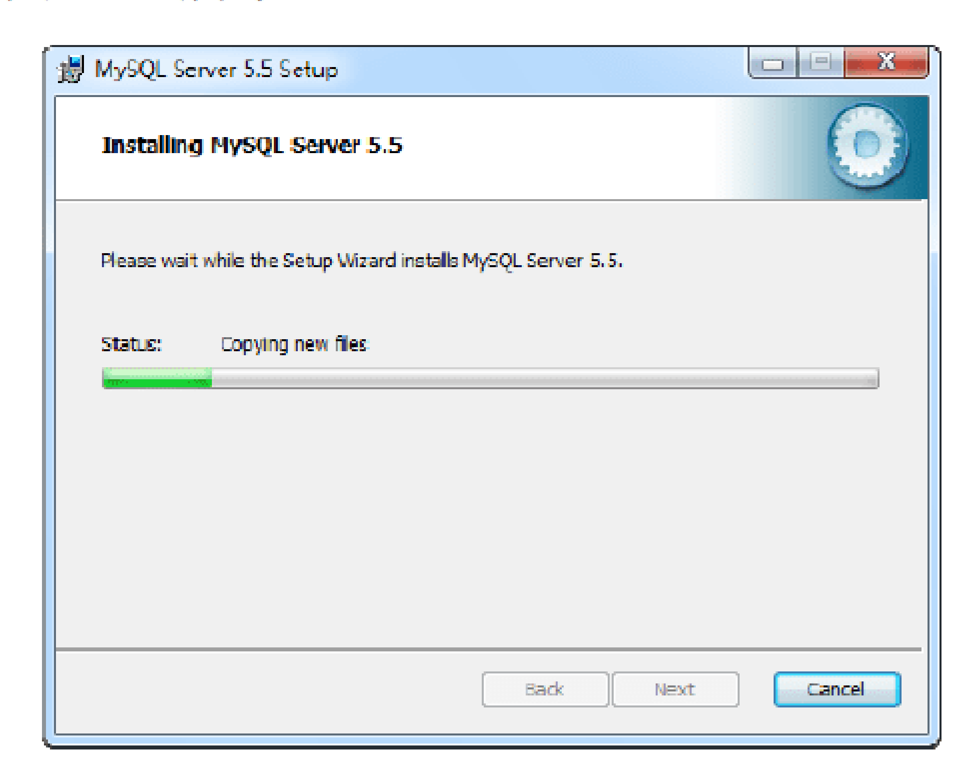


图 1.36　MySQL 安装进度对话框





Note

（7）在安装完成后会弹出如图 1.37 和图 1.38 所示的两个广告对话框，在广告对话框（一）中单击 Next 按钮。

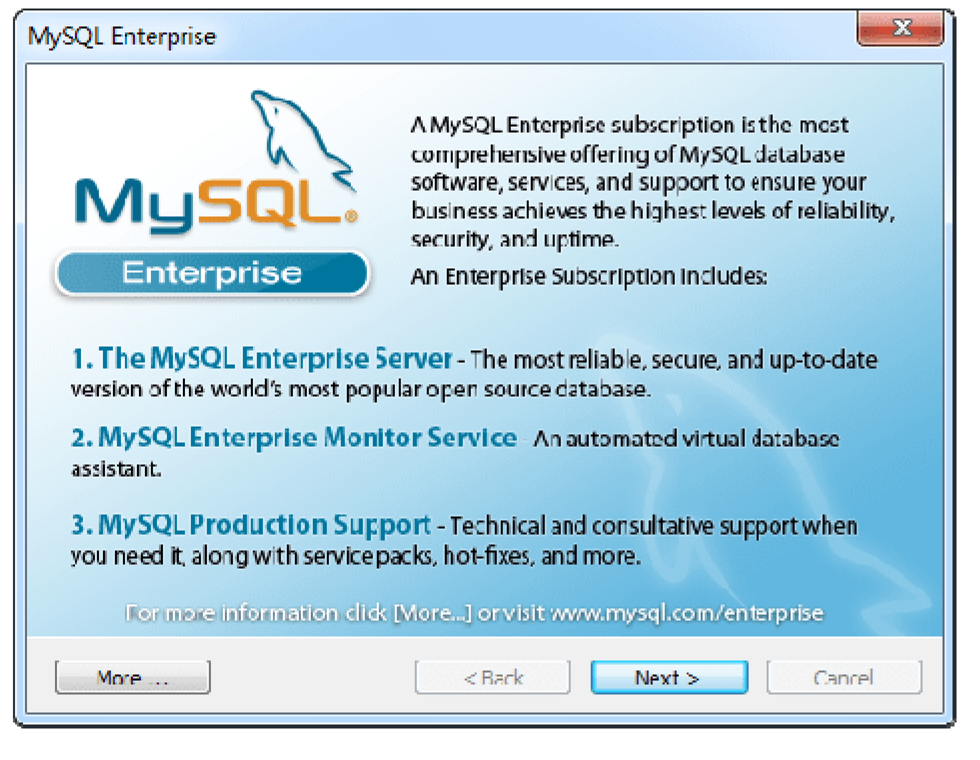


图 1.37　广告对话框（一）



图 1.38　广告对话框（二）

（8）在广告对话框（二）中单击 Next 按钮，弹出如图 1.39 所示的对话框。

（9）选中 Launch the MySQL Instance Configuration Wizard 复选框，单击 Finish 按钮，弹出如图 1.40 所示的对话框。

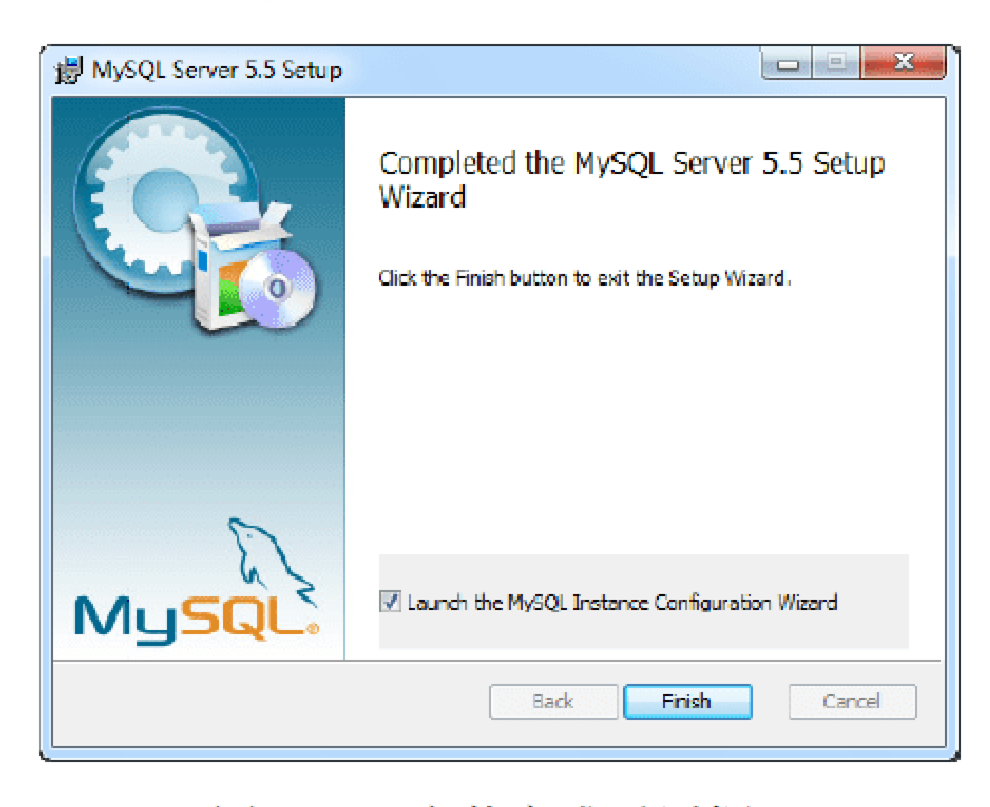


图 1.39　安装完成对话框

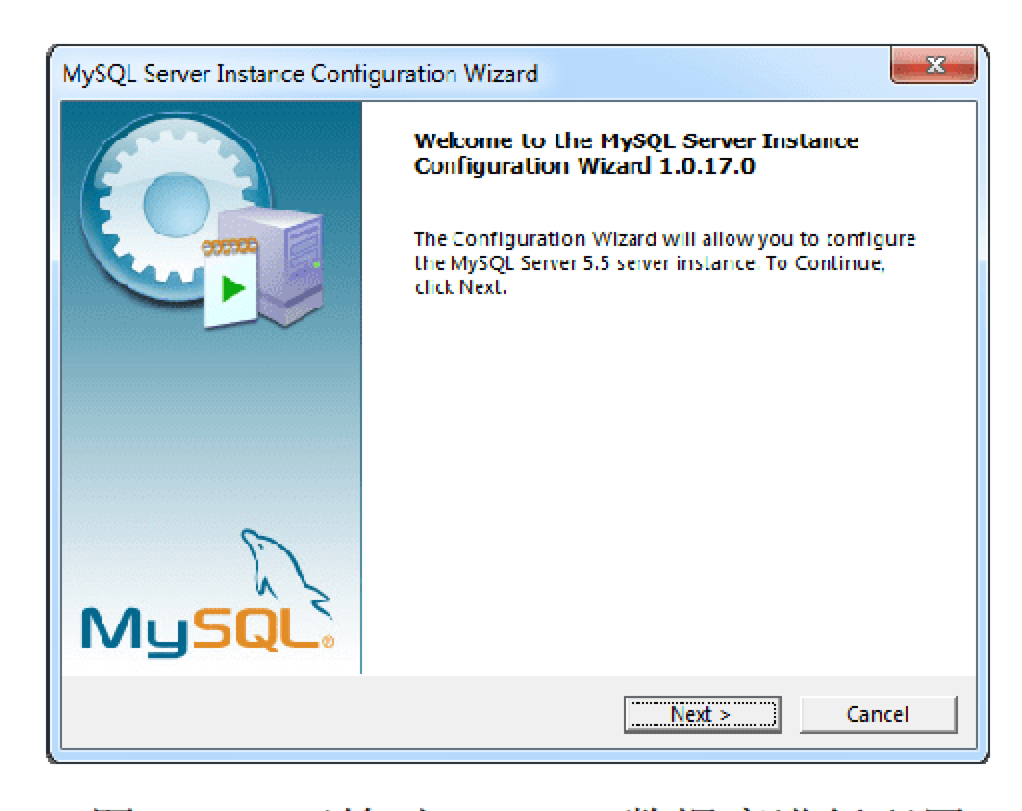


图 1.40　开始对 MySQL 数据库进行配置

（10）单击 Next 按钮，显示如图 1.41 所示的对话框。

（11）选中 Detailed Configuration（详细配置）单选按钮，单击 Next 按钮，显示如图 1.42 所示的对话框。

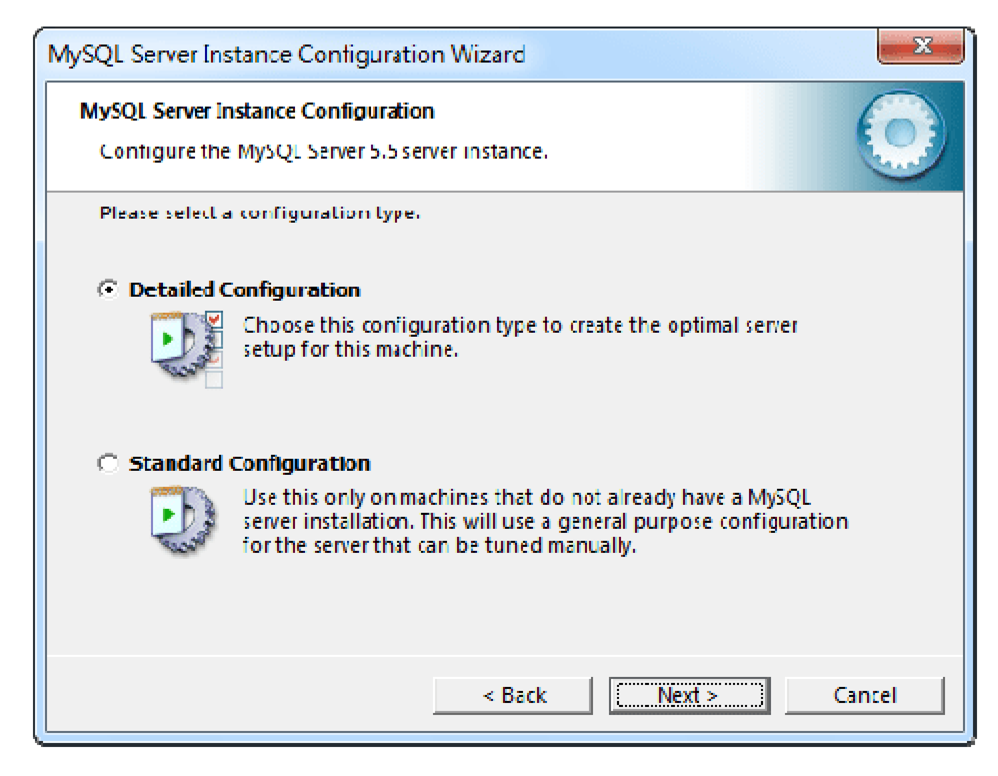


图 1.41　选择使用哪种配置方式

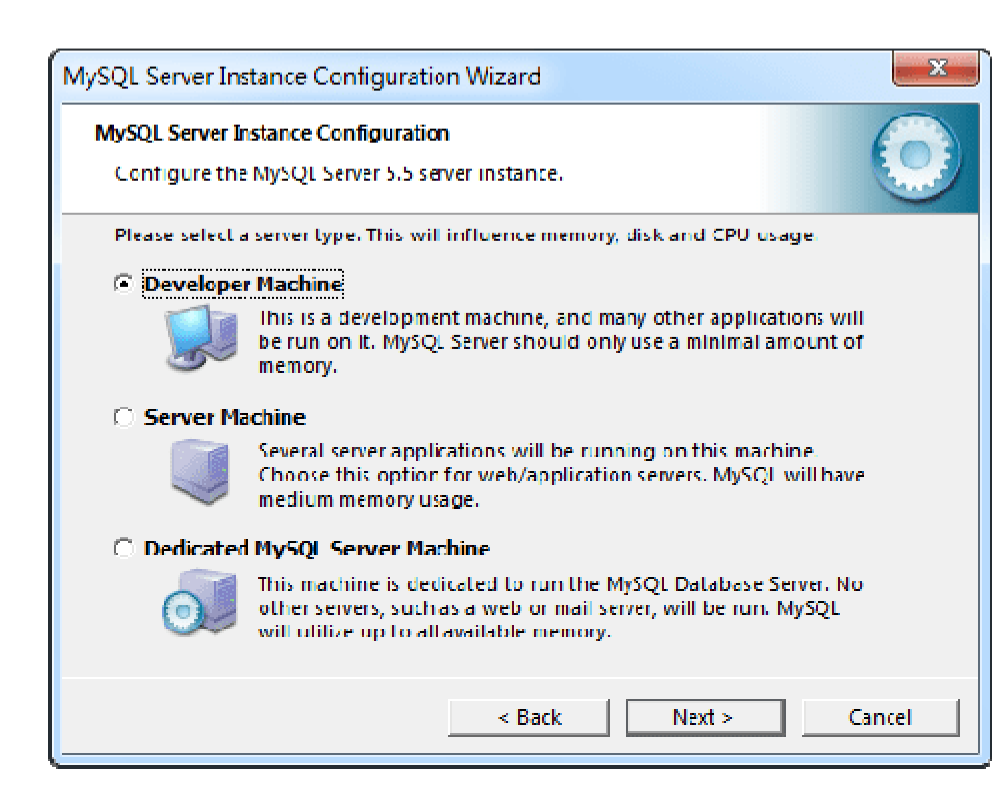
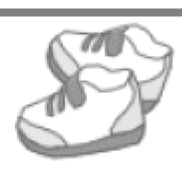


图 1.42　选择服务器类型





（12）选中 Developer Machine（开发者机器）单选按钮，单击 Next 按钮，打开如图 1.43 所示的对话框。

（13）选中 Multifunction Database（多功能数据库）单选按钮，单击 Next 按钮，打开如图 1.44 所示的对话框。

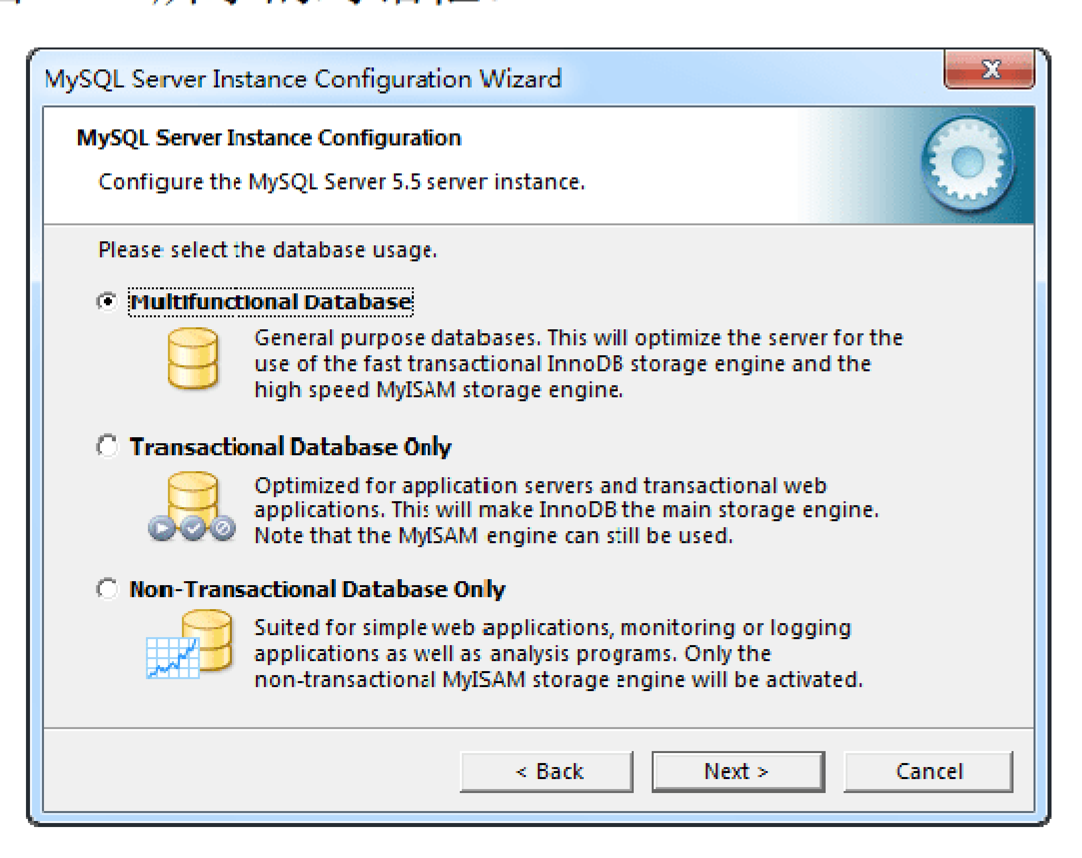


图 1.43　选择数据库类型

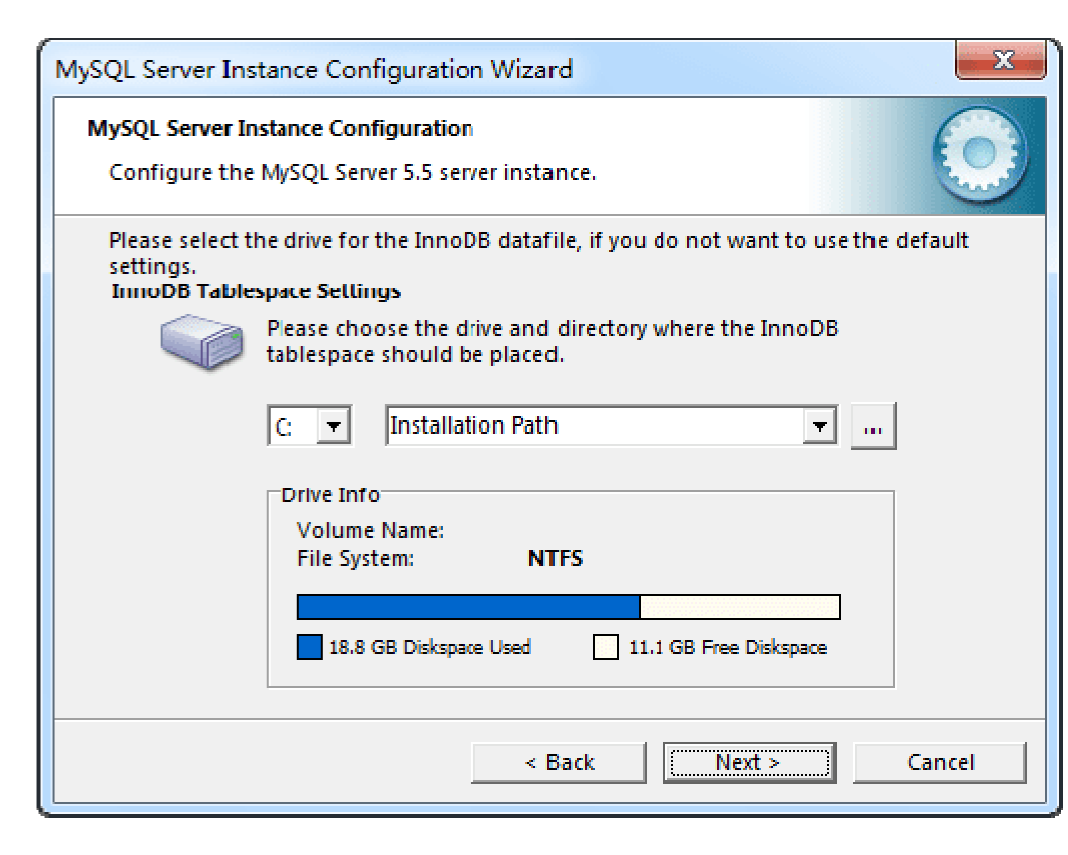


图 1.44　选择 InnoDB 表空间保存位置

（14）使用默认设置，单击 Next 按钮，打开如图 1.45 所示的对话框。

（15）使用默认设置，单击 Next 按钮，打开如图 1.46 所示的对话框。

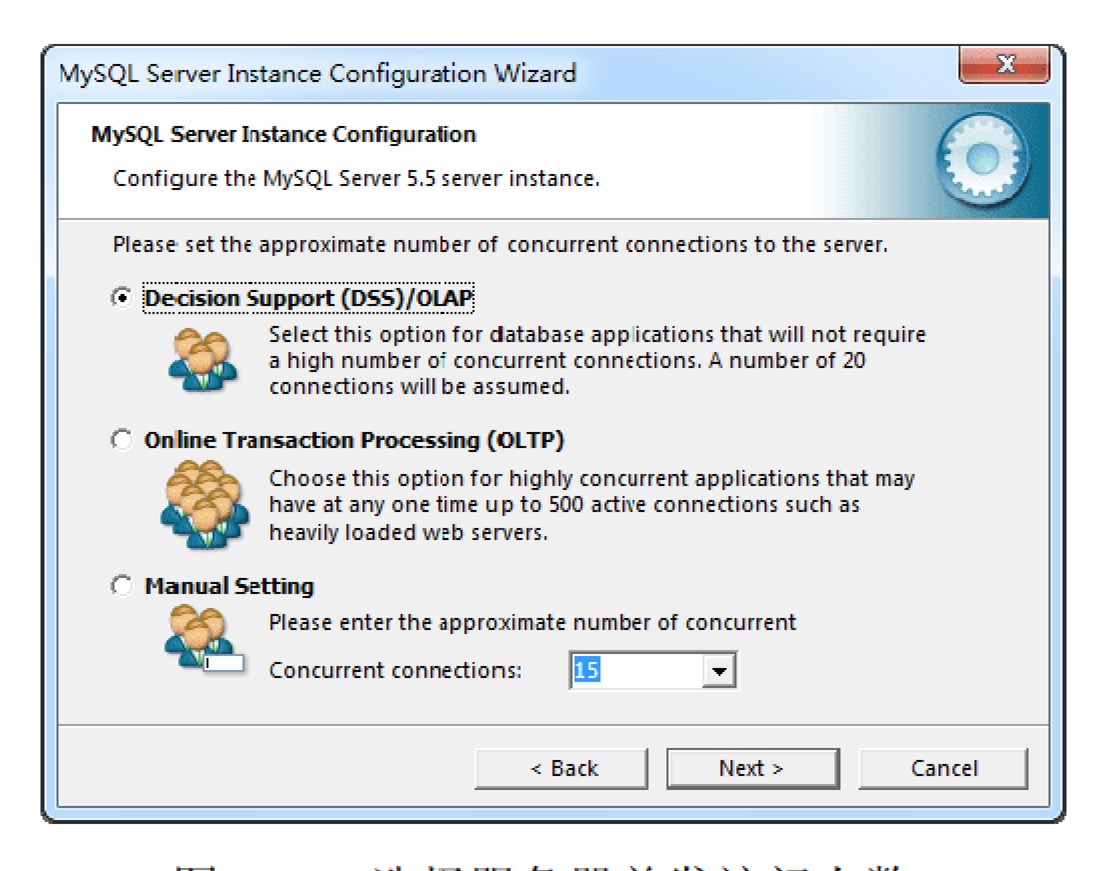




图 1.45　选择服务器并发访问人数

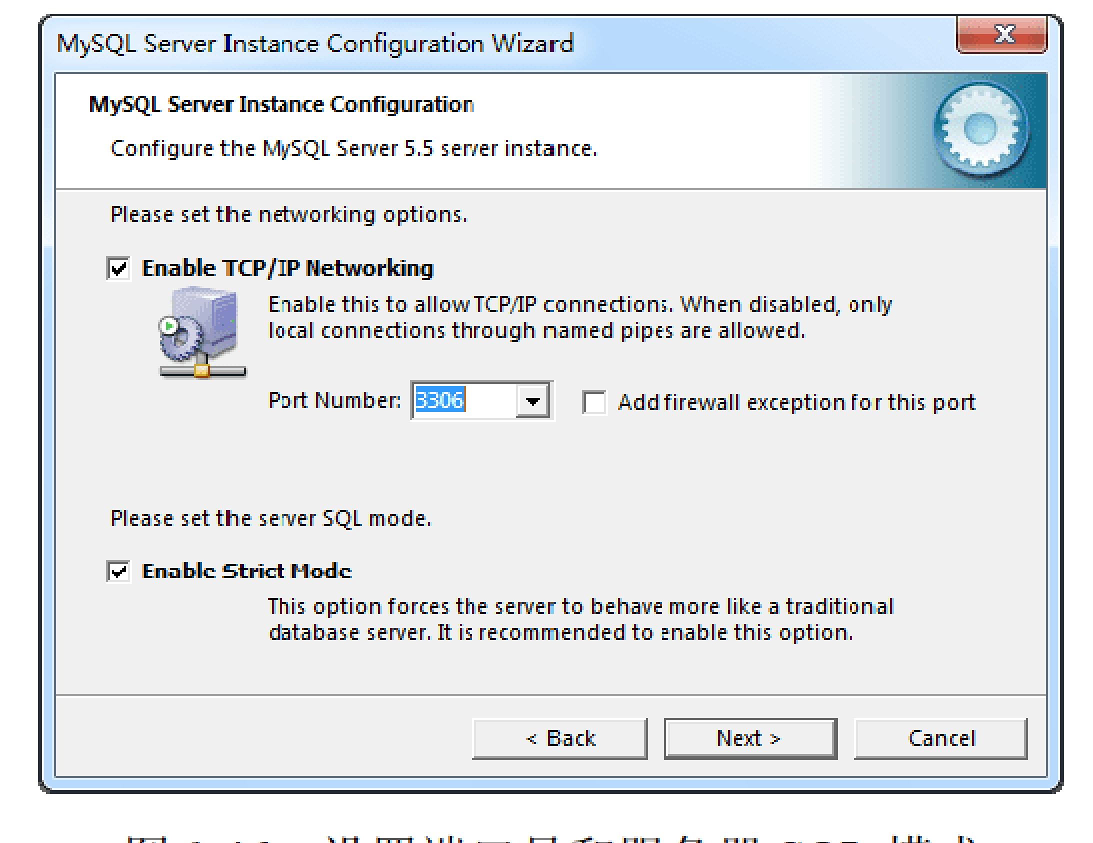


图 1.46　设置端口号和服务器 SQL 模式

注意：

MySQL 使用的默认端口是 3306，在安装时也可以修改为其他的端口号，如 3307。但是一般情况下，不要修改默认的端口号，除非 3306 端口已经被占用。

（16）使用默认设置，单击 Next 按钮，打开如图 1.47 所示的对话框。

（17）选中 Manual Selected Default Character Set/Collation 单选按钮，设置字符集编码为 utf8，单击 Next 按钮，进入如图 1.48 所示的对话框。

（18）选中 Install As Windows Service 和 Include Bin Directory in Windows PATH 复选框，单击 Next 按钮，进入如图 1.49 所示的对话框。

（19）输入数据库的密码“111”，单击 Next 按钮，打开如图 1.50 所示的对话框。



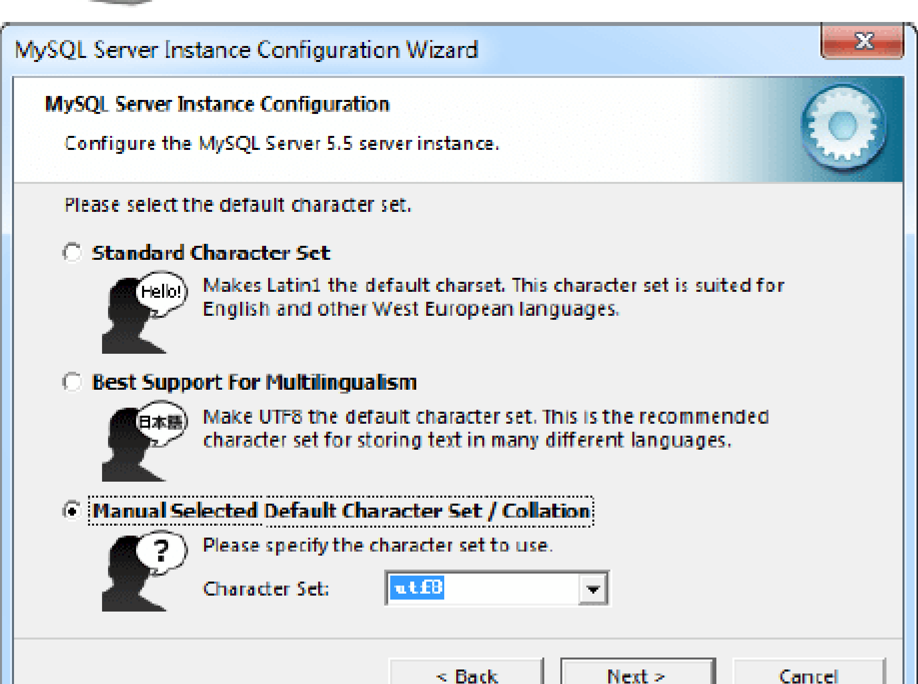


图 1.47 设置默认的字符集

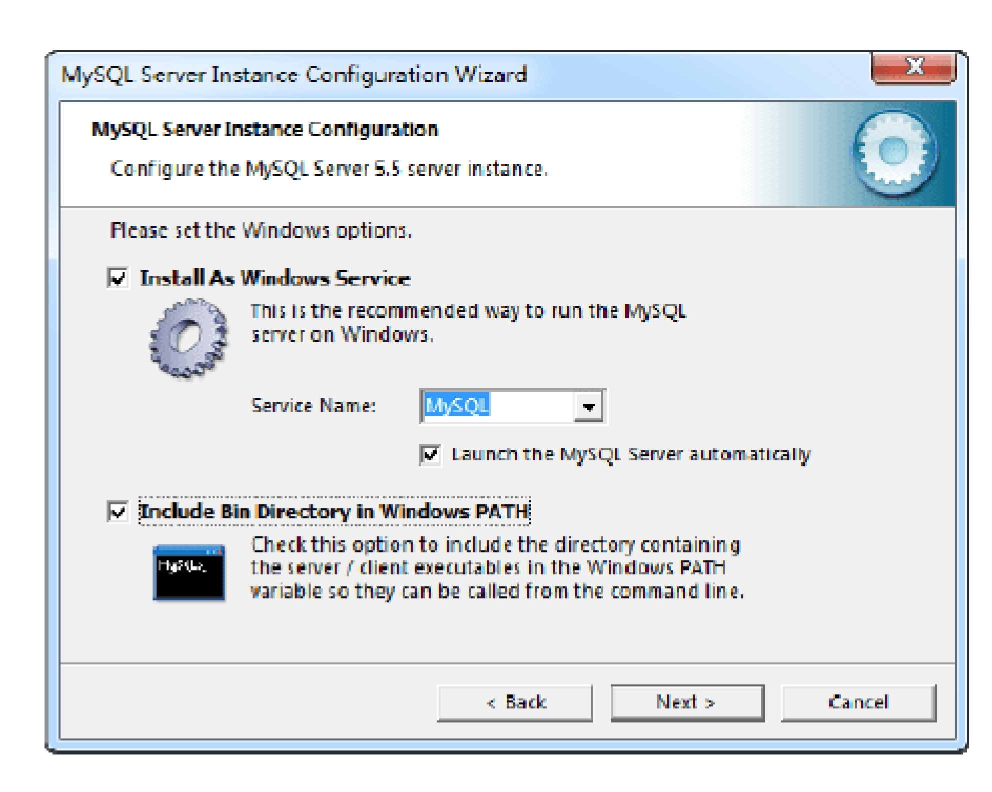


图 1.48 针对 Windows 系统进行的设置

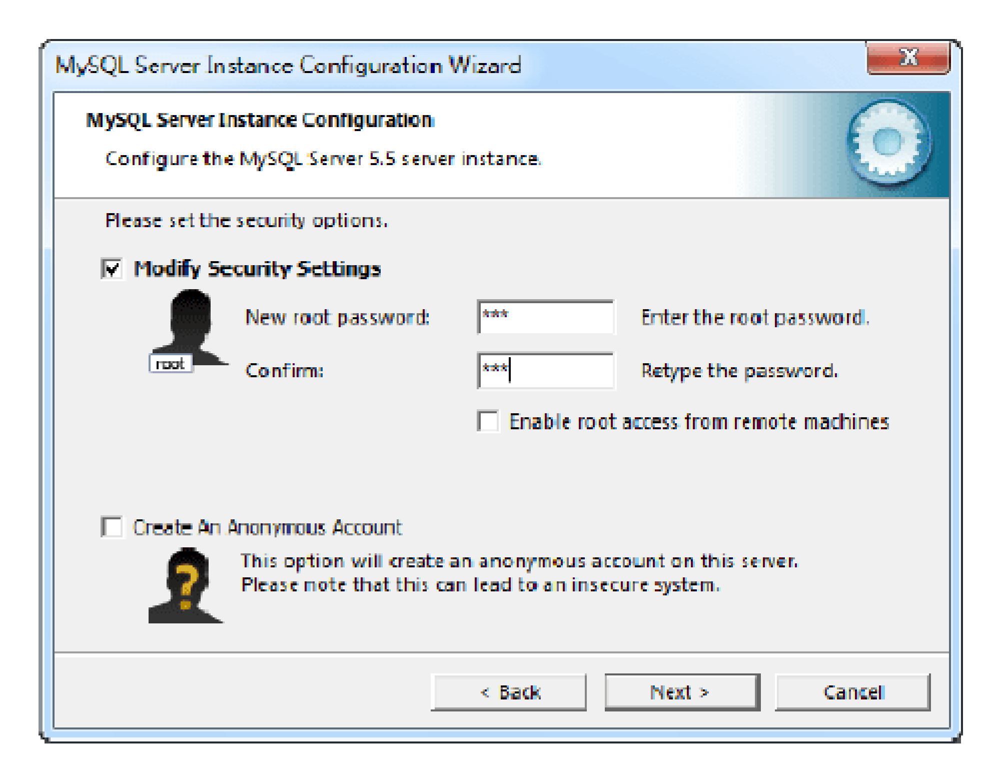


图 1.49 输入数据库的密码

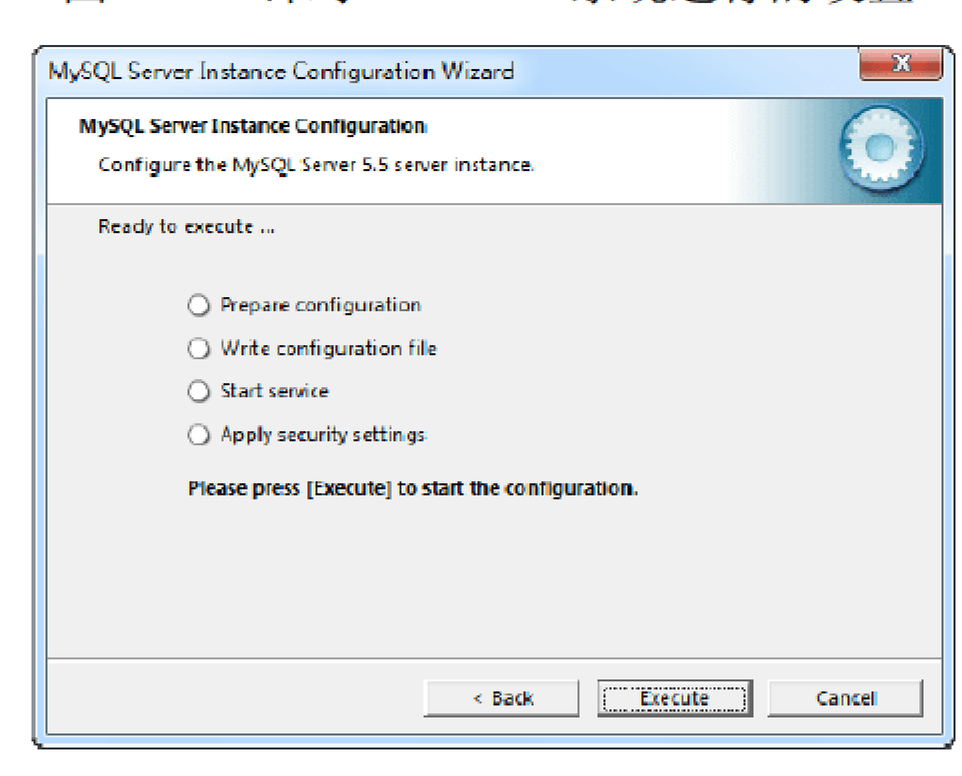


图 1.50 确认配置对话框

注意：

在安装 MySQL 数据库时，一定要牢记在上述步骤中设置的默认用户 root 的密码，这是在访问 MySQL 数据库时必须使用的。

（20）单击 Execute 按钮，执行前面进行的各项配置，过程如图 1.51 所示。配置完成后的界面如图 1.52 所示。

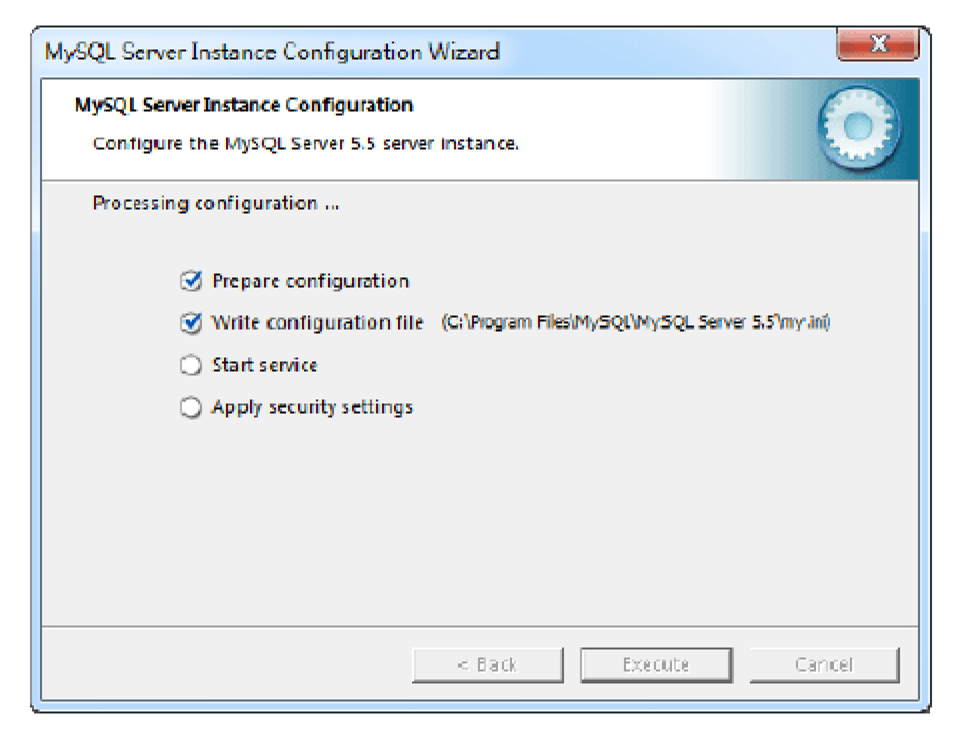


图 1.51 显示配置进度

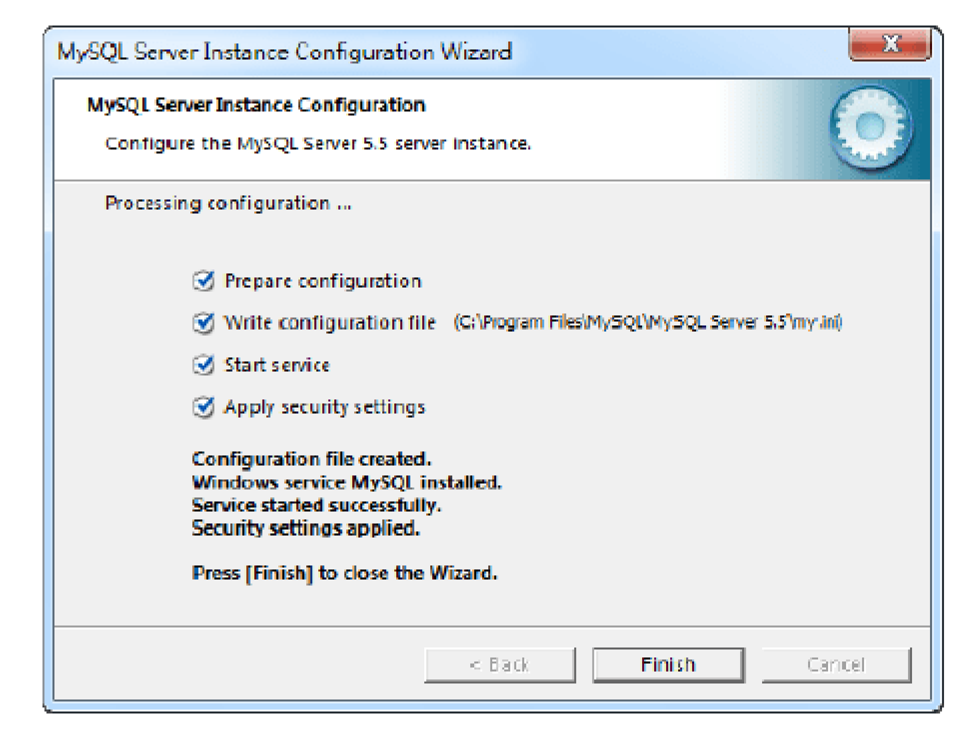
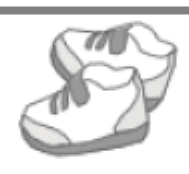


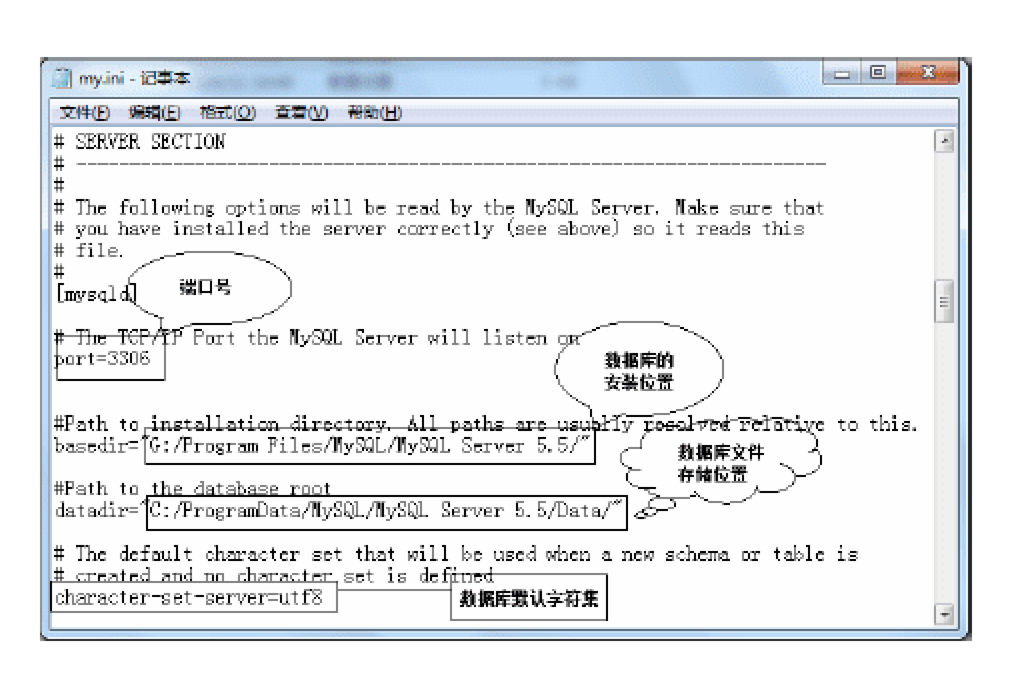
图 1.52 完成配置

至此，MySQL 安装成功。如果要查看 MySQL 的安装配置信息，则可以通过 MySQL 安装目录下的 my.ini 文件来完成。

在 my.ini 文件中可以查看到 MySQL 服务器的端口号、MySQL 在本机的安装位置、MySQL 数据库文件存储的位置以及 MySQL 数据库的编码等配置信息。参考内容如图 1.53 所示。







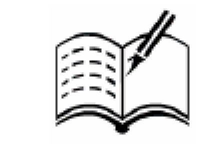


图 1.53　my.ini 文件的配置信息

专家点评

（1）如果在机器中同时安装了 IIS 和 Apache，则应对 IIS 或 Apache 的端口号进行修改，以避免这两个服务器发生冲突。

（2）如果本机使用 IIS 服务器配置 PHP 的开发环境，那么就不要再使用集成化的安装包来配置。因为它们会发生冲突，从而导致配置不能生效。

（3）在安装 PHP 的过程中应注意，一定要将 PHP 文件夹下的 libMySQL.dll 复制到系统根目录下的 Windows/system32 文件夹下。

（4）IIS 服务器的伪静态设置（UrlRewrite 地址变换功能）。

通过地址变换可以实现页面静态功能，方便搜索引擎收录，例如，http://***/news.php?id=11 可以变为 htt://***/news-11.html，同时还隐藏了开发标记，避免每个页面都使用.php 等后缀，提高了网站的安全性。

注意，IIS 默认是不支持这个功能的，需要单独安装插件。其具体操作步骤如下：

①进入 http://www.helicontech.com/download-isapi_rewrite3.htm 网站。

②下载 ISAPI_Rewrite 3 Lite installation package（精简版）。

③安装 ISAPI_Rewrite3_0048_Lite.msi。

④打开 IIS，在 Web 属性的“ISAPI 筛选器”中添加位于安装目录的 Helicon\ISAPI_Rewrite3 文件夹中的 ISAPI_Rewrite.dll，名称自行设置。

至此，IIS 的插件安装配置成功。运行位于安装目录的 Helicon\ISAPI_Rewrite3 文件夹中的 Helicon Manager.exe，编辑正则表达式，实现对 URL 地址的转换。同时将文件保存在实例根目录下，并命名为“.htaccess”。

问题 4　PHP 集成开发环境的特点有哪些？

问题阐述

PHP 集成开发环境软件有很多，如 XAMPP、EasyPHP 和 Apperv 等，那么每个集成开发环境都有哪些特点呢？

专家解答

1. XAMPP

XAMPP 是一个功能全面的 AMPP（Apache、MySQL、PHP、Perl）软件包，这是 Linux



平台上可以使用的几种非商业 XAMPP 中间件之一。采用这种紧密的集成，XAMPP 可以运行任何程序：从个人主页到功能全面的产品站点（注意，这仅仅用于开发目的，出于安全考虑，XAMPP 还不适于用在产品服务器上）。



2. EasyPHP

EasyPHP 是一个 Windows 下的 Apache+Mysql+Perl/PHP/Python 开发包，包中集成了 PHP、Apache 和 MySQL，同时也集成了一些辅助的开发工具，如数据库管理工具，phpMyAdmin 和 php 调试工具 Xdebug，无须配置即可运行。EasyPHP 是由法国人开发，经过 EasyPHP 整合后的 Apache、MySQL 及 PHP 精简很多，运行速度比独立安装的 Apache、MySQL 及 PHP 相对较快且较为稳定。目前最新版本是 EasyPHP 5.3.6.0、PHP 5.3.6 VC9、Apache 2.2.17 VC9、MySQL 5.5.10、Xdebug。

3. AppServ

如果用户的本地机器没有安装过 PHP、MySQL 等系统，那么使用该软件则可以帮助用户迅速搭建完整的底层环境。

专家点评

基于上述 3 个集成的开发环境安装软件，笔者认为还是使用 AppServ 要好一些，因为使用 XAMPP 和 EasyPHP，在应用 jpgraph 开发图像时会出现一些错误。而使用 AppServ 就不会有任何的问题。建议版本为 AppServ 5.10。

对于 PHP 开发环境的搭建，最好的方法还是个人搭建，搭建的操作步骤在网上完全可以找到，只是在搭建成功后需要对其进行一些修改，并开启一些 PHP.ini 的设置。例如：

```
display_error=ON
```

输出错误信息。但是如果是在互联网的服务器中使用，那么就要关闭该属性。

```
log_errors=on
```

在日志文件中输出错误信息。

```
error_reporting = E_ALL & ~E_NOTICE
```

设置错误级别。

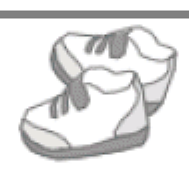
问题 5　如何应用 AppServ 搭建 PHP 开发环境?

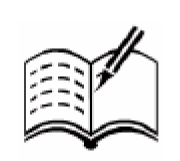
问题阐述

AppServ 将 Apache、PHP、MySQL 和 phpMyAdmin 等服务器软件和工具安装配置完成后打包处理。开发人员只要到网站上下载该软件后安装，即可完成 PHP 开发环境的快速搭建。这种搭建 PHP 开发环境的方法非常适合初学者，那么应用 AppServ 搭建 PHP 开发环境的步骤有哪些呢?

专家解答

应用 AppServ 集成化安装包搭建 PHP 开发环境的操作步骤如下：





（1）双击 AppServ-win32-2.5.7.exe 文件，打开如图 1.54 所示的 AppServ 启动页面。

（2）单击 Next 按钮，打开如图 1.55 所示的 AppServ 安装协议页面。

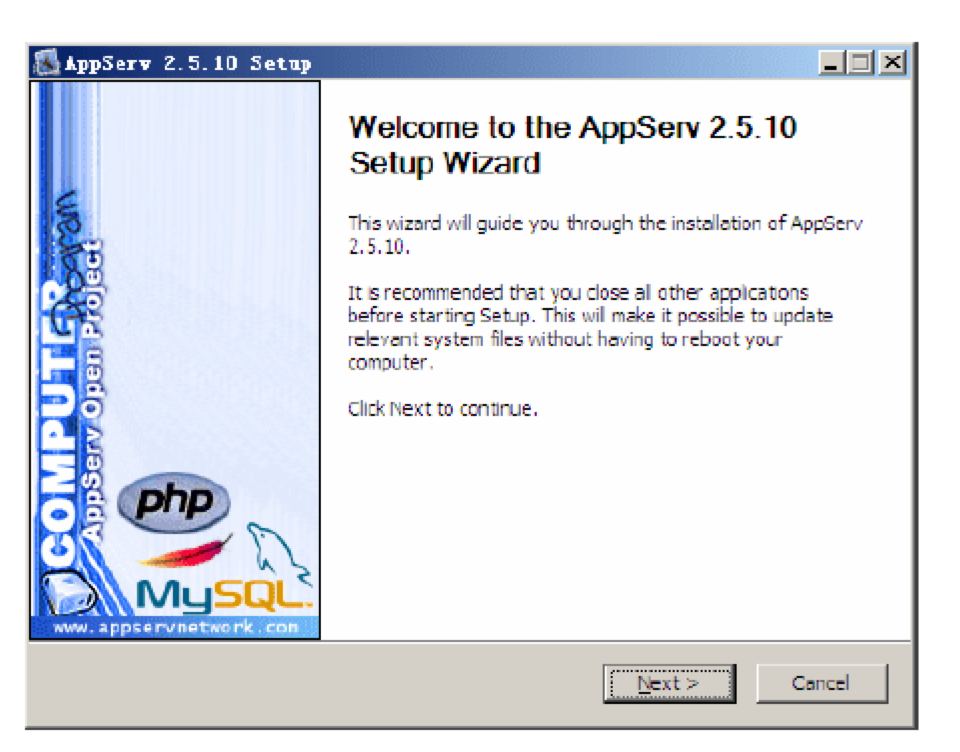


图 1.54　AppServ 启动页面

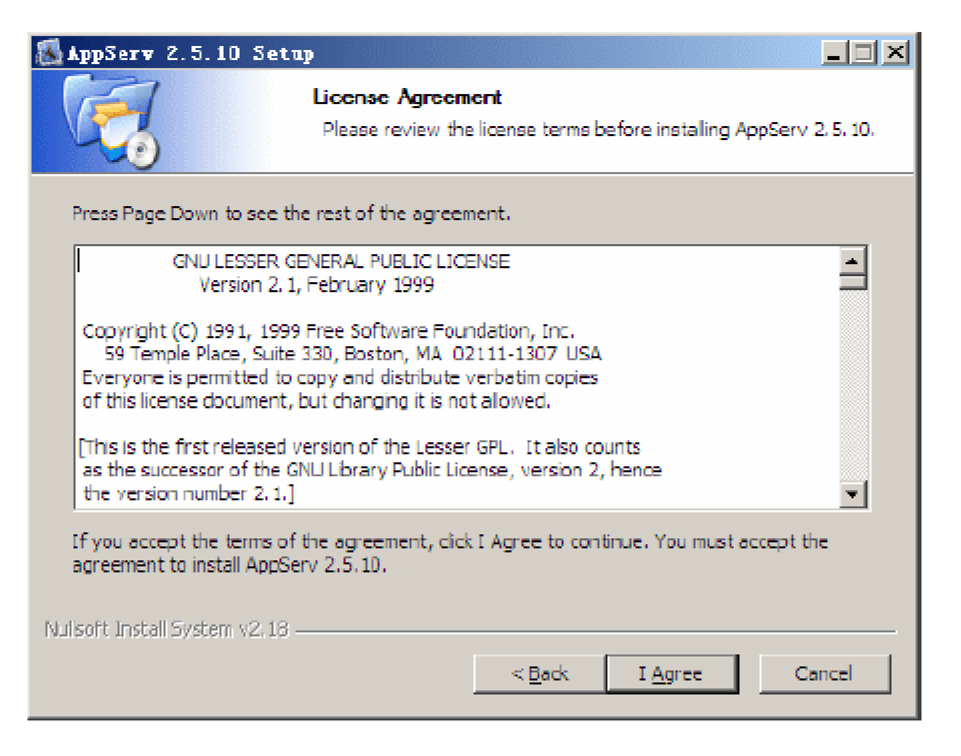


图 1.55　AppServ 安装协议

（3）单击 I Agree 按钮，打开如图 1.56 所示的页面。在该页面中可以设置 AppServ 的安装路径（默认安装路径一般为：C:\AppServ），安装完成后，Apache、MySQL、PHP 都将以子目录的形式存储到该目录下。

（4）单击 Next 按钮将打开如图 1.57 所示的页面。在该页面中可以选择要安装的程序和组件（默认为全选状态）。

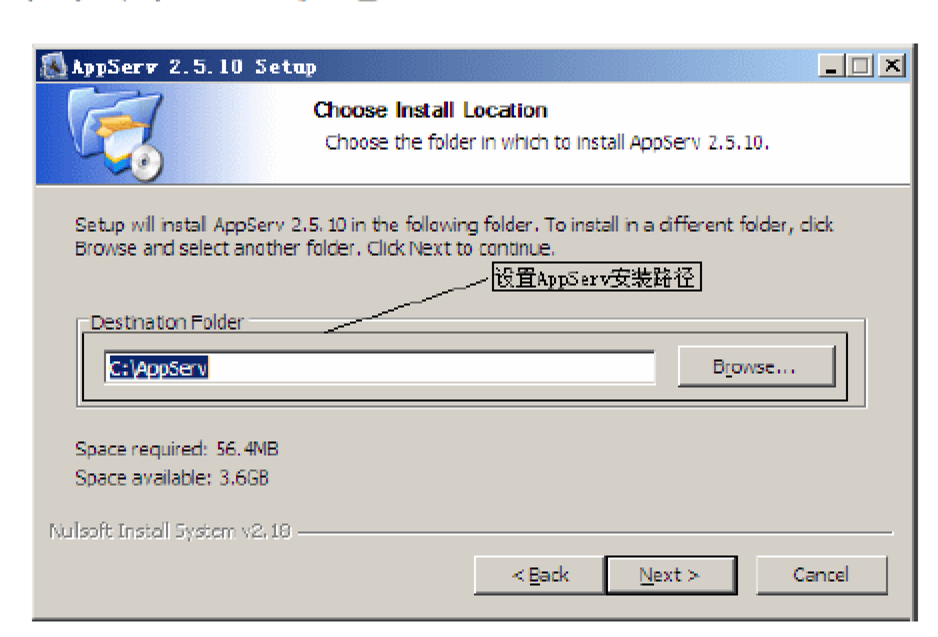


图 1.56　AppServ 安装路径选择

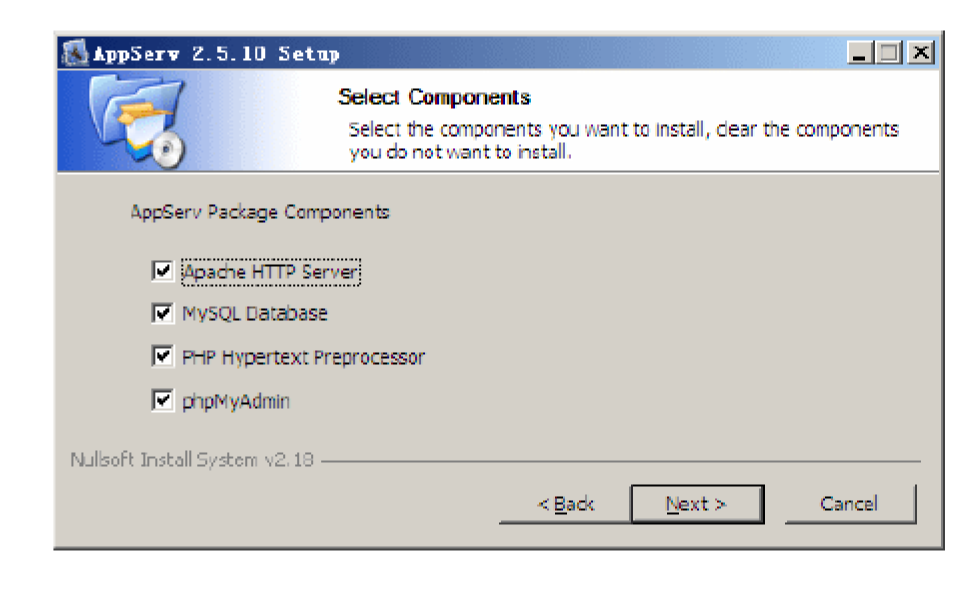


图 1.57　AppServ 安装选项

（5）单击 Next 按钮，打开如图 1.58 所示的页面，该页面主要用于设置 Apache 的端口号。

（6）单击 Next 按钮，打开如图 1.59 所示的页面，该页面主要用于对 MySQL 数据库的 root 用户的登录密码及字符集进行设置，这里将字符集设置为“GB2312 Simplified Chinese”，表示 MySQL 数据库的字符集将采用简体中文形式。

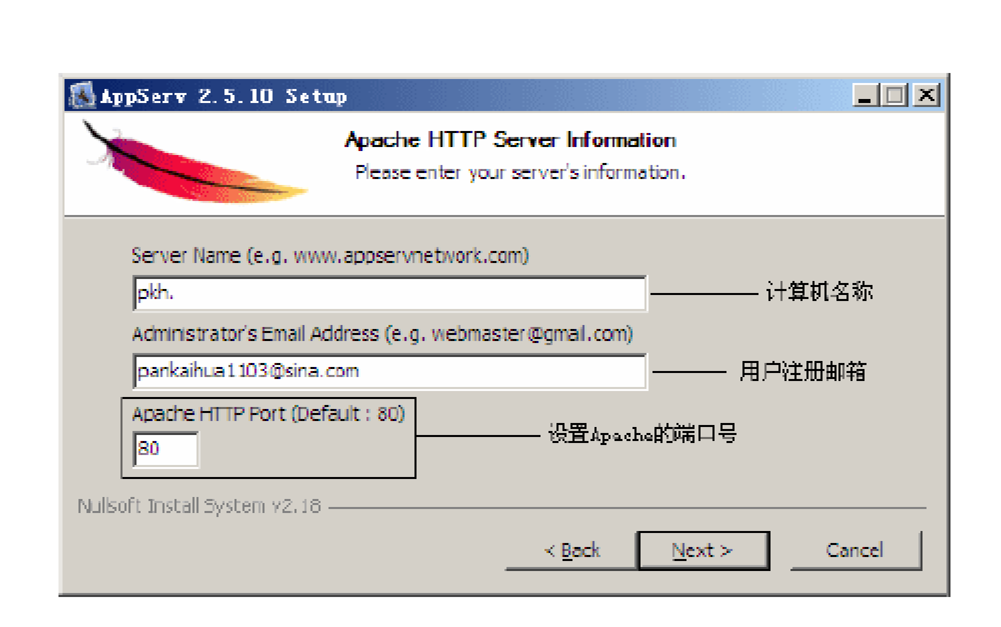


图 1.58　Apache 端口号设置

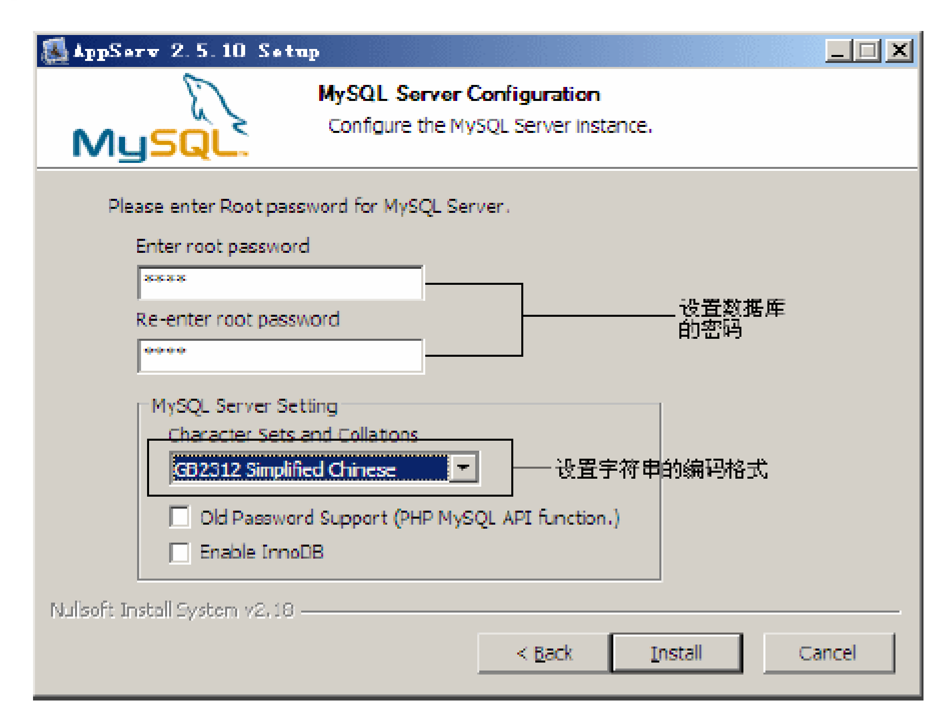
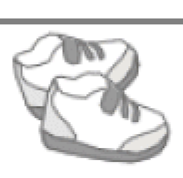


图 1.59　MySQL 设置



（7）单击 Install 按钮开始安装，如图 1.60 所示。

（8）安装完成后可以在开始菜单的 AppServ 相关操作列表中启动 Apache 及 MySQL 服务，如图 1.61 所示。





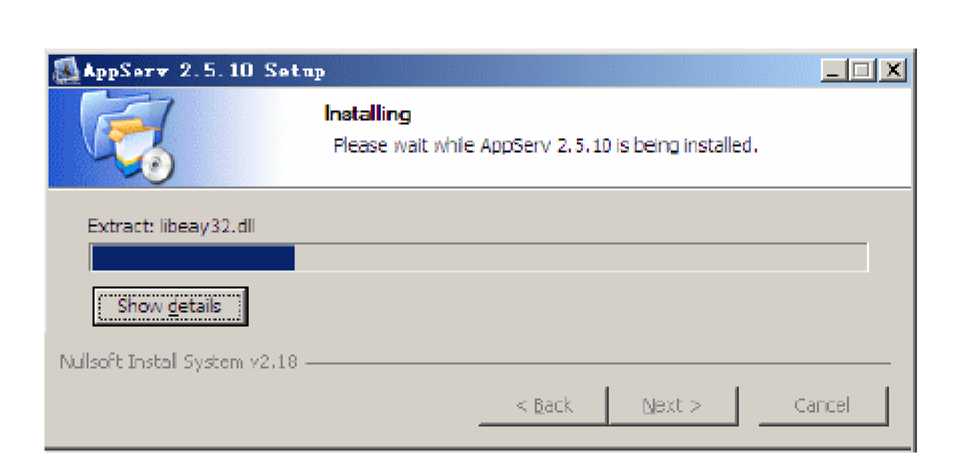


图 1.60　AppServ 安装页面



图 1.61　AppServ 安装完成页面

（9）安装好 AppServ 之后，整个目录默认安装在 C:\AppServ 下。此目录下包含 4 个子目录，如图 1.62 所示，用户可以将所有网页文件存放到“www”目录下。

（10）打开 IE 浏览器，在地址栏中输入“http://localhost/”或“http://127.0.0.1/”，如果打开如图 1.63 所示的页面，则说明 AppServ 安装成功。

AppServ
Apache2.2 —— Apache的存储目录
MySQL —— MySQL的存储目录
php5 —— PHP的存储路径
www —— 网页文件及phpMyAdmin的存储路径

图 1.62　AppServ 目录结构



图 1.63　AppServ 测试页

（11）编写一个 PHP 脚本，代码如下：

```
<?php
  echo "欢迎进入 PHP 的世界！";
?>
```

将该文件命名为 text.php，保存到 AppServ 的 www 文件夹下。然后在 IE 浏览器中输出 http://127.0.0.1/text.php。运行结果如图 1.64 所示。

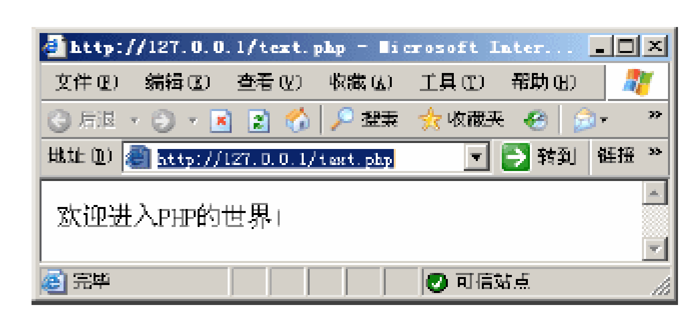
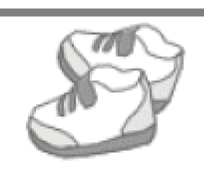
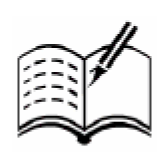


图 1.64　PHP 程序运行结果

注意：

在使用 Apache 搭建 PHP 开发环境时，必须确保在系统中没有安装 Apache、PHP 和 MySQL。否则，要先将这些软件卸载，然后才能应用 Apache。





专家点评

在安装过程中有以下几个步骤需要注意。

（1）AppServ 安装路径的选择。具体安装在哪个磁盘下用户可以自行选择，也可以使用默认的安装位置。

（2）在填写服务器信息时，需要注意 Server Name 中填写的是计算机名称；Administrator Email Address 中填写的是用户的邮箱地址；最关键的是 Apache HTTP Port，它设置的是服务器的端口，默认使用 80，读者也可以更改该端口号。

注意：

如果用户使用默认的 80 端口，那么必须确定本机上的 80 端口未被占用。如果用户机器上已安装了 IIS，那么 80 端口被 IIS 占用了，在安装 AppServ 时就不能再使用 80 端口了。建议用户在安装 AppServ 时使用默认 80 端口，如果 IIS 占用了 80 端口，用户可以将 IIS 端口修改为 82 或者其他。当然，如果不想修改 IIS 的端口，也可以直接将 AppServ 默认的 80 端口修改为其他。

（3）在 MySQL 数据库的设置中，应注意 MySQL 数据库的密码和字符集的设置。这里设置的密码是用户在以后通过 phpMyAdmin 访问数据库时需要使用的，所以用户应牢记设置的这个密码。同时该密码也是程序中连接数据库时所用的密码。至于编码格式，建议用户使用 GB2312 Simplified Chinese，也就是简体中文。

问题 6　如何通过 XAMPP 配置 PHP 开发环境？

问题阐述

XAMPP 是一个易于安装且包含 MySQL、PHP 和 Perl 的 Apache 发行版，那么如何通过安装 XAMPP 配置 PHP 开发环境呢？

专家解答

XAMPP 1.7.3 版的具体安装步骤如下：

（1）运行.exe 安装文件，将弹出如图 1.65 所示的操作界面。在这里设置 XAMPP 的安装路径。

（2）安装路径设置完成后，单击 Install 按钮，执行 XAMPP 的安装，如图 1.66 所示。

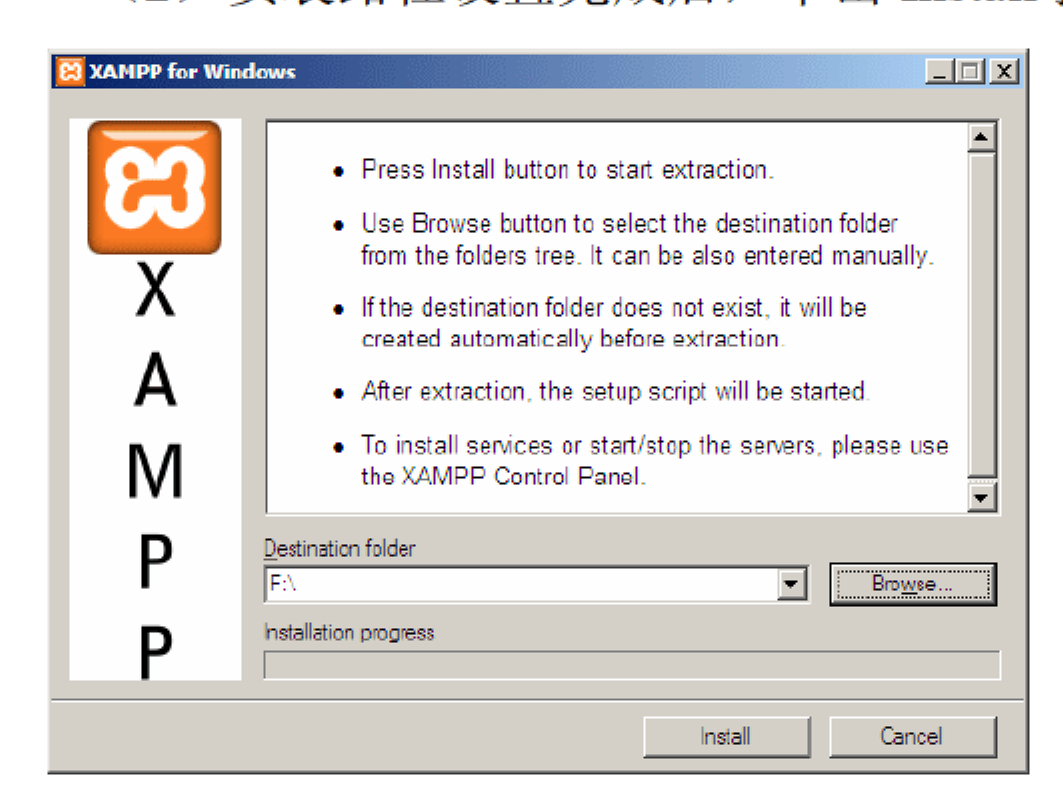


图 1.65　设置 XAMPP 的安装路径

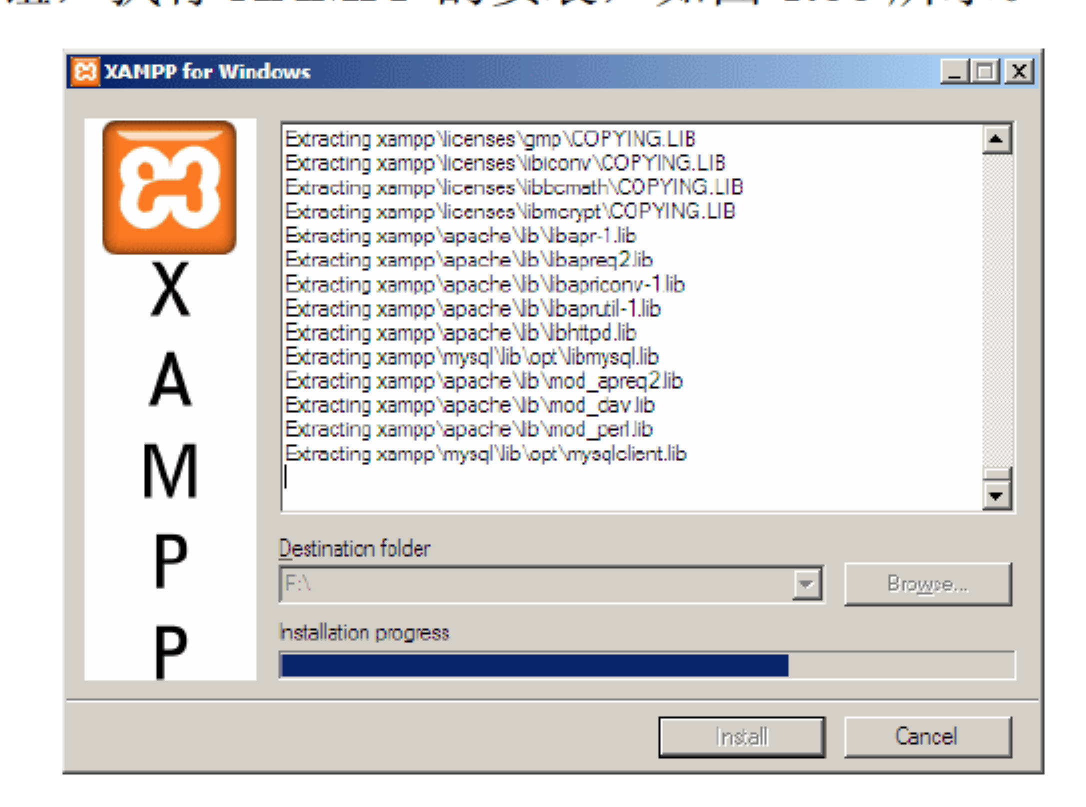


图 1.66　安装 XAMPP





Note

（3）XAMPP 安装成功后，将弹出如图 1.67 所示的操作界面。在该界面中对 XAMPP 进行设置，包括是否创建桌面快捷方式及是否在开机时启动 XAMPP 等。

```
C:\WINDOWS\system32\cmd.exe

################################################################################
# XAMPP 1.7.3 - Setup                                                          #
#------------------------------------------------------------------------------#
# Copyright 2009 Carsten Wiedmann (FreeBSD License)                            #
#------------------------------------------------------------------------------#
# Authors: Carsten Wiedmann <carsten_sttgt@gmx.de>                             #
#          Kay Vogelgesang <kvo@apachefriends.org>                             #
################################################################################

1. start XAMPP Control Panel
2. relocate XAMPP
   (current path: F:\xampp)
3. disable HTTPS (SSL)
4. disable Server Side Includes (SSI)
5. enable IPv4 only (current: IPv4/6 (auto))
6. disable mod_perl
7. disable Apache::ASP

x  Exit
```

图 1.67　设置 XAMPP

至此，XAMPP 安装、配置完毕。

专家点评

下面对 XAMPP 中的配置文件进行简单介绍。

XAMPP 中各种配置文件的存储位置如下。

- ☑ Apache 基本配置：.\xampp\apache\conf\httpd.conf
- ☑ Apache SSL：.\xampp\apache\conf\ssl.conf
- ☑ Apache Perl（仅限插件）：.\xampp\apache\conf\perl.conf
- ☑ PHP：.\xampp\php\php.ini
- ☑ MySQL：.\xampp\mysql\bin\my.ini
- ☑ phpMyAdmin：.\xampp\phpMyAdmin\config.inc.php
- ☑ FileZilla FTP 服务器：.\xampp\FileZillaFTP\FileZilla

问题 7　XAMPP——Linux 版 PHP 集成化安装包

问题阐述

在 Linux 操作系统下通过 XAMPP 配置 PHP 的开发环境的步骤有哪些?

专家解答

在 Linux 操作系统下，使用集成软件 XAMPP 的 Linux 版来配置 PHP 开发环境。Linux 下安装 XAMPP 的步骤如下：

（1）在 Linux 操作系统下，选择“主菜单”→“系统工具”→“终端”命令。

（2）在命令模式下，首先进入系统的根目录下。

（3）通过 mkdir 命令在根目录下创建一个 opt 目录。

（4）通过 tar xvfz 命令将 XAMPP 解压缩到 opt 目录下。

注意：

在 Linux 操作系统下执行 XAMPP 的解压缩时，必须进入系统的根目录下，并且要在根目录下创建一个子目录，然后才可以执行解压缩的命令。否则，将提示无法找到指定的目录或文件。



（5）按回车键，执行 XAMPP 的解压缩，直到安装成功。其具体使用的命令如图 1.68 所示。

（6）安装成功后查看/opt/lampp 目录，如图 1.69 所示。



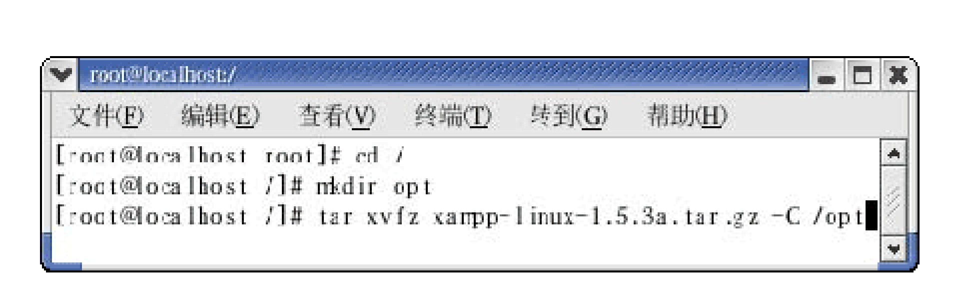


图 1.68　XAMPP 的安装

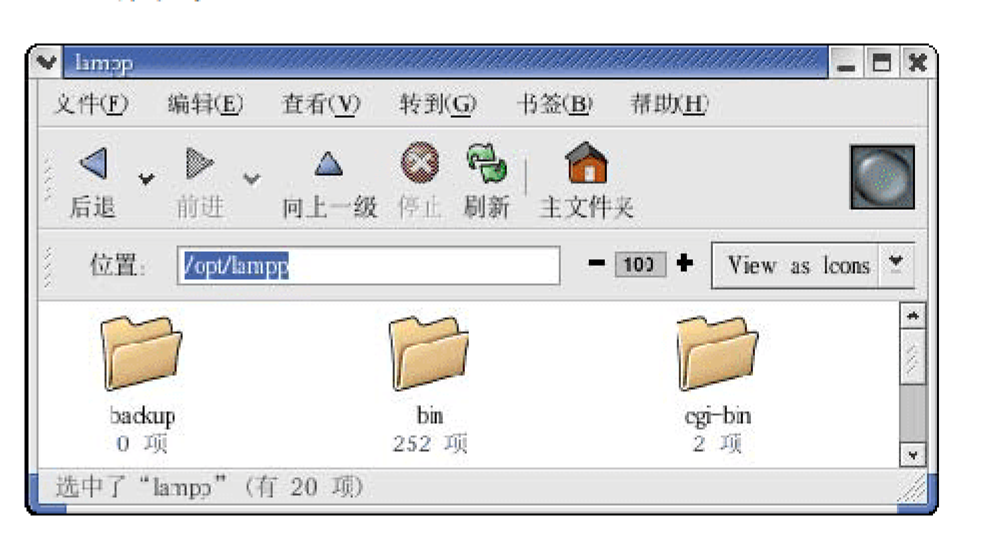


图 1.69　查看安装的内容

（7）在 Mozilla 浏览器中输入“http://127.0.0.1/index.php”，运行结果如图 1.70 所示。

注意：

在 Linux 操作系统下安装 XAMPP，必须在 Linux “主菜单”→“系统工具”下的 “终端”命令中通过指令来完成，切记不要使用任何微软操作系统下的工具来进行操作。Linux 下卸载 XAMPP 的命令：rm –rf /opt/lampp。

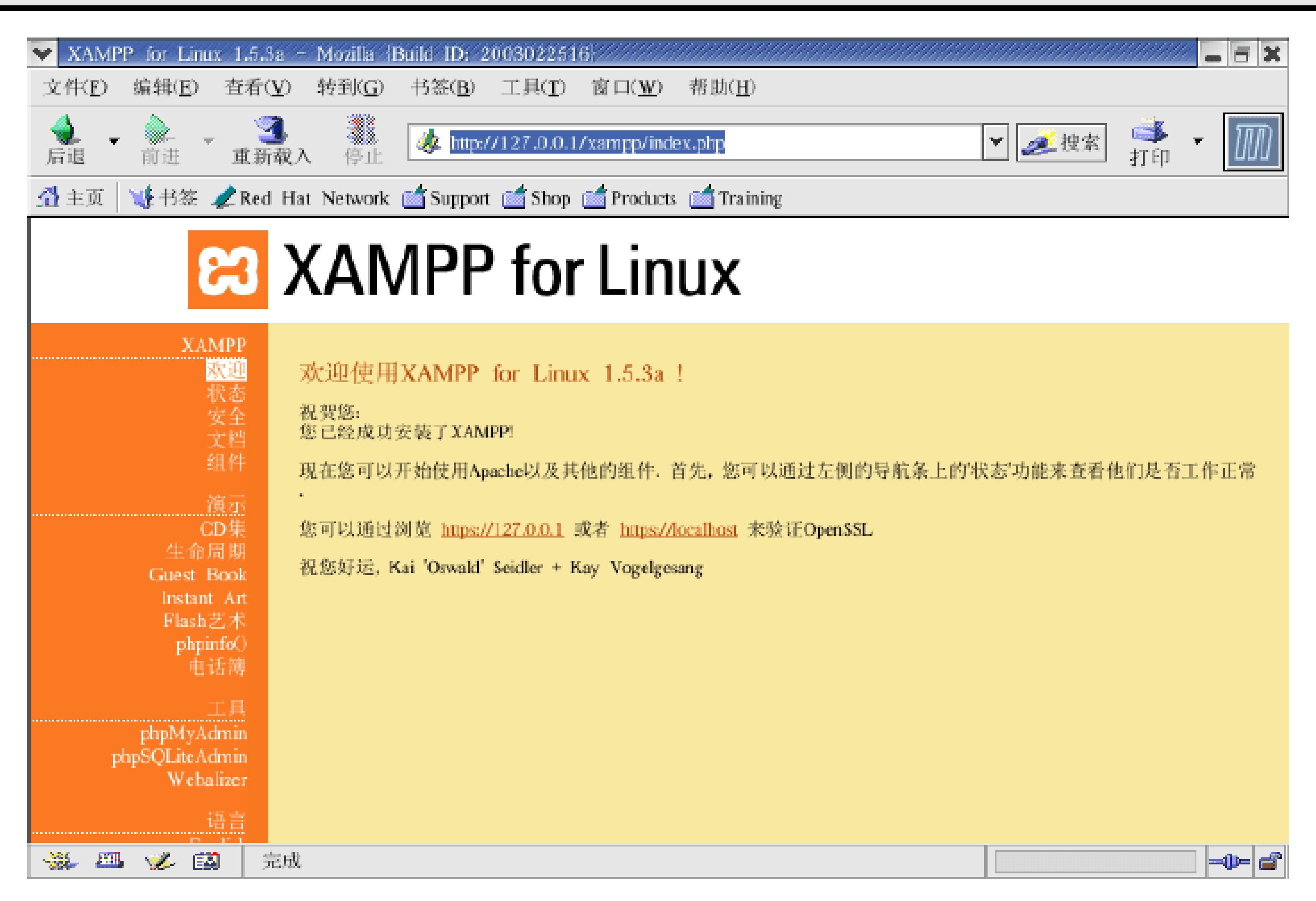


图 1.70　Linux 版的 XAMPP

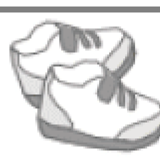
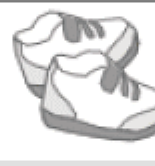
问题 8　Apache 配置文件全解

问题阐述

PHP 开发环境配置成功后，有关 Apache 服务器的配置文件（httpd.conf）被存储于 Apache2.2\conf\目录下，那么通过该配置文件都可以进行哪些操作呢？

专家解答

httpd.conf 存储于 Apache2.2\conf\目录下，是 Apache 服务器的配置文件，在该文件中可以修改 Apache 服务器的端口号、根目录，激活伪静态功能，以及配置 Apache 服务器的虚拟主机。配置文件的关键内容如下：

Note

```
# 设置 Apache 服务器安装文件的存储位置
ServerRoot "E:/wamp/apache2"
# 设置 Apache 服务器的端口号
Listen 80
# 开启伪静态功能
LoadModule rewrite_module modules/mod_rewrite.so
# 配置 PHP
LoadModule php5_module "E:/wamp/php/php5apache2_2.dll"
# 设置端口号
ServerName localhost:80
# 定义 Apache 服务器的 Web 存储路径
DocumentRoot "E:/wamp/www"
#
<Directory />
    Options FollowSymLinks
    AllowOverride All
    Order deny,allow
    Deny from all
    Satisfy all
</Directory>
# 定义 Apache 服务器的 Web 存储路径
<Directory "E:/wamp/www">
    Options Indexes FollowSymLinks
    Order Deny,Allow
    Deny from all
    Allow from 127.0.0.1
</Directory>
# 设置伪静态
<Directory "E:/wamp/apache2/cgi-bin">
    AllowOverride All
    Options All
    Order allow,deny
    Allow from all
</Directory>
#
```

设置 Apache 服务器的虚拟主机，只需要在此文件的最后添加配置方法即可。

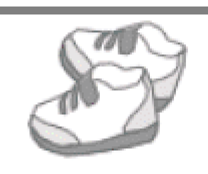
专家点评

在配置文件中，上述罗列的配置项都可以进行修改，进而达到读者想要的效果，切记在完成对配置文件的修改后应保存文件，并且重新启动 Apache 服务器，这样修改才会生效。

问题 9　PHP.INI 配置文件全解

问题阐述

PHP 开发环境配置成功后，在 Linux 操作系统中 php.ini 存储于/opt/lampp/etc/php.ini 文件夹下，而在 Windows 操作系统中 php.ini 存储于系统盘的 Windows 文件夹下，如果使用集成化的安装包来配置 PHP 开发环境，也有将 php.ini 存储于 PHP 安装文件夹下的。那么通过该配置





文件都可以进行哪些操作呢？

专家解答

php.ini 文件是 PHP 在启动时自动读取的配置文件，它是一个 ASCLL 文本文件，分为多个部分，每一部分包括相关的参数。每一部分的名称位于最前面的方括号内，接着是名称对数字，且每一名称都独占一行。使用规则 PHP 代码，对参数名称非常敏感，不能包含空格，但是参数可以是数字、字符串或布尔逻辑数。分号位于每一行的开始，作为指定标记，这就使选择使用或不使用 PHP 的这些特性变得很方便，而无须通过删除该行来实现。对某特性进行注释（即添加分号），则该行将不会被编译执行。每次修改完 php.ini 文件，必须重新启动 Apache 服务器，以使新的设置生效。

php.ini 是 PHP 的配置文件，用于加载各种函数库、设置错误级别和设置服务器的时间等。php.ini 文件的基本配置如表 1.1 所示。

表 1.1　php.ini 文件的基本配置

参　数	说　明	默　认　值
error_reporting	设置错误处理的级别。推荐值为E_ALL & ~E_NOTICE & ~E_STRICT，显示所有错误信息，除了提醒和编码标准化警告	E_ALL & ~E_NOTICE & ~E_STRICT
register_globals	通常情况下可以将此变量设置为Off，这样可以对通过表单进行的脚本攻击提供更为安全的防范措施	register_globals = On
include_path	设置PHP的搜索路径，该参数可以接收系列的目录。当PHP遇到没有路径的文件提示时，它将会自动检测这些目录，需要注意的是，当某些选项允许多个值时，应使用系统列表分隔符，在Windows下使用分号“；”，在Linux下使用冒号“：”	; UNIX: "/path1:/path2" ;include_path = ".:/php/includes" ; Windows: "\path1;\path2" ;include_path = ".;c:\php\includes"
extension_dir	指定PHP的动态链接扩展库的目录	\ext目录下
extension	指定PHP启动时所加载的动态链接扩展库。PHP的常用扩展库及其说明可参见表1.2	PHP的常用扩展库在初次安装配置后均被注释，需读者手动更改
file_uploads	设置是否允许通过HTTP上传文件	file_uploads=On
upload_tmp_dir	设置通过HTTP上传文件时的临时目录，如果为空，则使用系统的临时目录	upload_tmp_dir =空
upload_max_filesize	设置允许上传文件的大小，如“50MB”，必须填写单位	upload_max_filesize=2MB
post_max_size	控制在采用POST()方法进行一次表单提交中PHP所能够接收的最大容量。要上传更大的文件，则该值必须大于upload_max_filesize的值。 如upload_max_filesize=10MB，那么upload_max_filesize的值必须要大于10MB	post_max_size = 8MB
max_input_time	以秒为单位对通过POST、GET以及PUT方式接收数据时间进行限制	max_input_time = 60

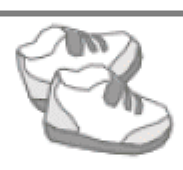
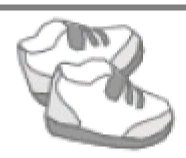




表 1.2　PHP 常用扩展库及其说明

扩　展　库	说　　明
php_ftp.dll	支持FTP函数库，可以实现客户机与服务器之间标准传送协议（FTP）
php_gd2.dll	支持图像处理函数库，支持.gif、.jpg、.png等多种图像格式
php_imap.dll	支持imap电子邮件处理函数库
php_mssql.dll	支持MsSQL数据库
php_msql.dll	支持mSQL数据库
php_mysql.dll	支持MySQL数据库
php_oracle.dll	支持Oracle数据库
php_pdf.dll	支持PDF文件处理函数库
php_sockets.dll	支持Sockets处理函数库
php_zlib.dll	支持zlib文件压缩函数库
php_pdo.dll	支持PDO数据库抽象层
php_pdo_mysql.dll	支持MySQL数据库
php_pdo_mssql.dll	支持MS SQL Server数据库
php_pdo_oci8.dll	支持Oracle数据库
php_pdo_odbc.dll	支持ODBC数据库
php_pdo_pgsql.dll	支持PGSQL数据库

问题 10　Apache 不能处理 PHP 文件

问题阐述

使用 RPM 安装了 PHP，但 Apache 却不能处理 PHP 文件，为什么？

专家解答

假设用户使用 RPM 正确安装了 PHP，则必须在 httpd.conf 文件中添加如下几行代码：

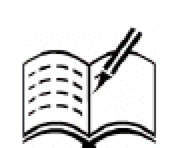
```
#Extra Modules
AddModule mod_php.c
AddModule mod_php3.c
AddModule mod_perl.c
#Extra Modules
LoadModule php_module modules/mod_php.so
LoadModule php3_module modules/libphp3.so
LoadModule perl_module modules/libperl.so
```

另外，再添加：

```
AddType application/x-httpd-php3.php3
```

到 httpd.conf 的全局部分，获得需要提供 PHP 支持的部分。





问题 11　无法找到 httpd.h 文件

问题阐述

PHP 和 Apache 一起编译时，显示无法找到 httpd.h 文件，但事实上该文件是存在的，这个问题怎么解决呢？

专家解答

用户需要让 PHP 配置即安装脚本知道自己的 Apache 源码的最高级目录，而不是包含 httpd.h 文件的目录。也就是说，应该指定"--with-apache=/path/to/apache/"而不是"--with-apache=/path/to/apache/src"。

问题 12　Apache 不能启动的解决方法

问题阐述

由于安装比较烦琐，有时出错在所难免，以下是几个在安装中出现的典型错误，仅供读者参考。

专家解答

（1）在安装 Web 服务器时提示“Socket error”或“address0.0.0.0：80 not available”。这个错误主要是由于当前环境中已经运行了其他 Web 服务器（如 IIS），80 端口已经被占用造成的。

解决方案：为 IIS 分配其他 IP 地址或停止，然后启动 Apache 的安装程序，选择“REPAIR”修复方式来重新安装。

（2）浏览器显示 400 的错误信息（或请求）。这是由于访问的.php 文件，PHP 引擎并没有找到，可能是在 httpd.conf 中设置的路径不正确造成的。

```
LoadModule php5_module "E:/wamp/php/php5apache2_2.dll"
```

解决方案：修改 httpd.con 配置文件，重新设置 PHP 引擎路径，保存后重启 Apache 服务器。

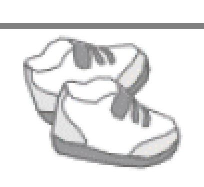
问题 13　如何解决 Apache 服务器端口冲突？

问题阐述

如何解决 Apache 服务器端口冲突？

专家解答

Apache 服务器的核心配置文件是 httpd.conf，要修改其端口号，自然也在该文件中进行。定位到 Listen 80 一行，修改 Listen 的值，实现 Apache 服务器端口号的更改。切记修改后应保存，并重新启动 Apache 服务器。







专家点评

IIS 服务器、迅雷的默认端口号为 80，与 Apache 服务器默认端口号相同。由于采用了相同的端口号 80，因此，在运行网页时就会发生冲突。

如果用户机器上安装了 IIS 服务器，就需要修改 IIS 的默认端口，否则将导致 Apache 服务器无法正常工作。更改 IIS 的默认侦听端口号 80，可以在 IIS 的管理器中进行设置或者停止 IIS 的服务。

另外，用户也可以在安装 Apache 服务器时将默认的端口号进行更改，从而解决两个服务器或与其他软件共用一个端口号而产生冲突的问题。如果对 Apache 的端口号进行了修改，那么在运行 PHP 项目时，输入的地址就会发生一些变化。

例如，若将 Apache 的端口号修改为 8080，那么在运行本机的 PHP 项目时，应输入“http://127.0.0.1:8080/”。

问题 14　如何配置 Apache 虚拟主机？

问题阐述

在配置 PHP 运行环境时，如果是作为服务器来使用，那么就少不了配置 Apache 的虚拟主机。在虚拟主机的配置中可以实现域名与指定项目的对接，在网页中输入指定的域名，即可访问对应的项目内容。配置 Apache 的虚拟主机是创建 PHP 服务器必须掌握的一项内容，那么该如何配置 Apache 虚拟主机呢？

专家解答

配置 Apache 虚拟主机，其前提是在作为服务器的计算机中已经正确地配置了一个 PHP 运行环境，配置 Apache 虚拟主机的操作在 Apache 的 httpd.conf 文件中进行，其具体步骤如下：

（1）定位到 Apache2\conf\ httpd.conf 文件，在此文件中添加如下内容：

```
<VirtualHost *:80>
    ServerAdmin any@any.com
    DocumentRoot "E:\wamp\www\mingribook"
    ServerName www.mingribook.com
    ErrorLog "logs/phpchina1.com-error.log"
    CustomLog "logs/phpchina1.com-access.log" common
</VirtualHost>
```

第 1 行，定义虚拟服务器的标签，指定端口号。

第 2 行，指定一个邮箱地址，可以随意指定。

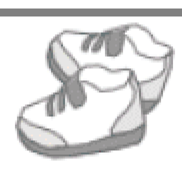
第 3 行，定义要访问的项目在 Apache 服务器中的具体路径。

第 4 行，指定服务器的访问名称，即与项目绑定的域名。

第 5、6 行，定义 Apache 中日志文件的存储位置。

第 7 行，定义虚拟服务器的结束标签。

上述 7 行代码即完成一个虚拟服务器的配置操作，如果存在多个域名，并且需要绑定 Apache 服务器下的多个项目，那么就以此类推，重复上述操作，为每个域名绑定不同的项目文件即可，即修改 DocumentRoot 和 ServerName 指定的值。



（2）在完成虚拟主机的配置之后，需要保存 httpd.conf 文件，且重新启动 Apache 服务器。

专家点评



如果在 Apache 服务器中配置了虚拟主机，那么在 Apache 服务器中将不能再直接通过路径的方式访问其他项目，而必须使用虚拟主机中指定的服务器名称才能访问。例如，定义 DocumentRoot 访问文件为"E:\wamp\www\mingribook"，ServerName 的名称为 127.0.0.1，那么在访问程序时可以直接在浏览器中输入 127.0.0.1，即可访问此项目的内容。但是，在 E:\wamp\www\文件夹下还存储了另外一个项目的文件夹 mrbccd，如果在没有设置虚拟主机的情况下，可以直接通过 http://127.0.0.1/mrbccd 的形式访问这个项目，此时再以这种形式进行访问时却不能正常输出。



问题 15　如何动态加载扩展库？

问题阐述

在 PHP 开发时，如果临时需要一个新扩展库，该如何加载呢？

专家解答

看下面的 php.ini 配置文件。

```
enable_dl=on
```

首先要确认上面的 enable_dl 的状态值为 on，然后在 PHP 程序汇总就可以使用下列方法。

```
dl("gnu_gettext.dll");    //Win32 环境下
dl("gnu_gettext.so");    //Win32 环境下
```

使用上面的语法就可以实现动态地按需调入扩展库，其调入的目录路径根据 php.ini 中设置的 extension_dir 装入，从而节约资源，提高程序效率。

如果不确定该 php 扩展库是否装入，可用以下代码进行处理。

```
if(extension_loaded("")!=true){
dl("gnu_gettext.so");
}
```

表示如果要调用的扩展未加载内存，则立即装入该扩展库。

问题 16　register_globals（注册为全局变量）的开启与关闭

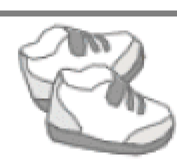
问题阐述

如何判断 register_globals 全局变量是开启还是关闭？

专家解答

register_globals 是 php.ini 中的一个配置，该配置直接影响 PHP 如何接收和传递参数。

如果它被设置为 Off，那么在获取通过 POST()或者 GET()方法传递的参数时，就需要使用



$_POST['user_name']或$_GET['user_name']。

如果它被设置为 On，那么就可以直接使用$user_name 来获取传递的值。

它的设置不只影响 form、url 参数的传递，同时也影响到了 SESSION 和 Cookie。同样，当它的值为 Off 时，获取 SESSION 和 Cookie 的值就需要使用$_SESSION[]和$_COOKIE[]；而当它的值为 On 时，就可以直接通过变量名称获取变量传递的值。



Note

技巧：

当用户遇到获取不到 form 表单中传递的值时，可以查看使用的方法是 POST 还是 GET，并且确认使用的变量名称是否正确。另外，可以查看一下 php.ini 文件，看 register_globals 是开启还是关闭，确定在获取参数值使用的方法有效。

专家点评

建议读者在配置 PHP 的开发环境时，将 register_globals 设置为 Off，有两个好处：第一可以提高代码的安全性；第二可以提高代码的可兼容性。

用户现在使用的所有 PHP 服务器中，register_globals 都是设置为 Off 的，所以一旦用户的程序在开发时设置的是 On，那么当它上传到服务器时，就会出现问题。在 PHP5 中，用户在配置时还有选择的余地，可以选择是 On 或 Off，但是在 PHP6 中这项内容已经被删除，所以用户必须做到未雨绸缪，养成一个良好的编程习惯。

问题 17　为什么要设置时区？

问题阐述

当安装完 PHP 开发环境后，获取系统时间时，会发现获取到的时间与系统时间不对应。那么如何对时区进行设置，并获取到相对应的时间呢？

专家解答

要获取本地当前的时间，则必须更改 PHP 语言中的时区设置。更改 PHP 语言中的时区设置有两种方法。

1. 在 php.ini 文件中设置时区

在 php.ini 文件中，定位到[date]下的";date.timezone ="选项，去掉前面的分号，并设置其值为当地所在时区使用的时间，修改内容如图 1.71 所示。



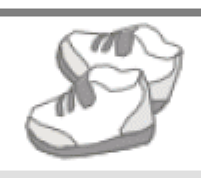
图 1.71　设置 PHP 的时区

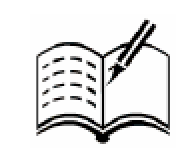
例如，如果当地所在时区为东八区，那么就可以设置"date.timezone ="的值为 PRC、Asia/Hong_Kong、Asia/Shanghai（上海）或 Asia/Urumqi（乌鲁木齐）等，这些都是东八区的时间。

设置完成后，保存文件，重新启动 Apache 服务器即可。

2. 通过 date_default_timezone_set 函数设置时区

在应用程序中，在日期、时间函数之前使用 date_default_timezone_set()函数就可以完成对时区的设置。date_default_timezone_set()函数的语法如下：






```
date_default_timezone_set(timezone);
```

参数 timezone 为 PHP 可识别的时区名称，如果时区名称 PHP 无法识别，则系统采用的是 UTC 时区。

例如，修改时区前获取到的时间如图 1.72 所示，修改时区后获取到的时间如图 1.73 所示。

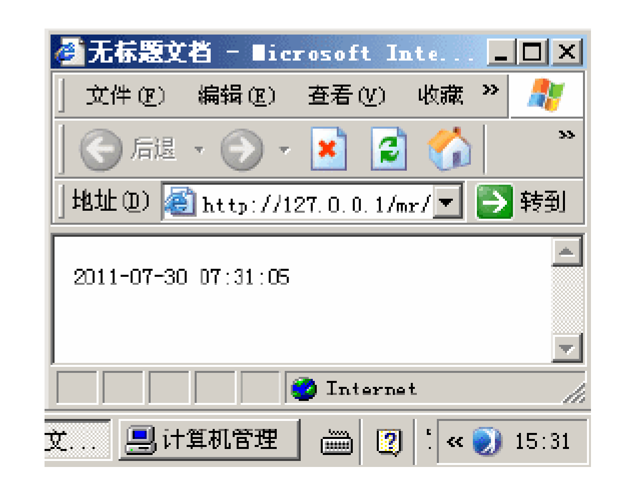


图 1.72　修改时区前获取的时间

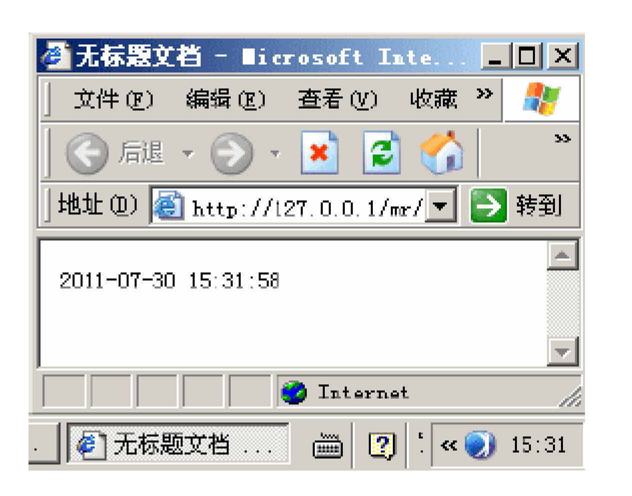


图 1.73　修改时区后获取的时间

获取时间的代码如下：

```
<?php
echo date("Y-m-d H:i:s");
?>
```

专家点评

在 PHP 语言中，日期、时间函数依赖于服务器的地区设置，而 PHP 默认设置的是标准的格林威治时间（即采用的是零时区），所以，如果用户没有对 PHP 的时区进行设置，那么使用日期、时间函数获取的将是英国伦敦本地时间（即零时区的时间）。

这也就是为什么要对 PHP 的时区进行设置的原因。例如，以东八区为例，如果当地使用的是北京时间，那么如果没有对 PHP 的时区进行设置，获取的时间就将比当地的北京时间少 8 个小时。

问题 18　如何在 Dreamweaver 中创建站点？

问题阐述

如何通过 Dreamweaver 开发工具创建一个站点呢？

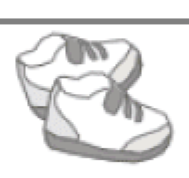
专家解答

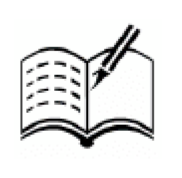
在 Dreamweaver 开发工具中，如果创建了站点，就不必在 IE 浏览器中输入地址，而只需直接按下键盘中的 F12 键，即可完成对所创建程序的浏览操作。

在 Dreamweaver 中创建站点和配置测试服务器时，一定要注意将本地的 HTTP 地址与测试服务器中的 URL 前缀统一，即都指定到站点的根目录下。例如，指定 HTTP 地址是“http://localhost//MR/Instance/”，那么测试服务器的 URL 前缀也必须是“http://localhost//MR/Instance/”或“http://127.0.0.1//MR/Instance/”。

在 Dreamweaver 中创建站点的操作步骤如下：

（1）打开 Dreamweaver 开发工具，选择菜单栏中的“站点”→“新建站点”命令，在弹出的对话框中输入站点名称，如图 1.74 所示。





（2）单击“高级”按钮，将弹出如图 1.75 所示的对话框。在该对话框中设置本地根文件夹，链接相对于“站点根目录”，并且设置 HTTP 地址。

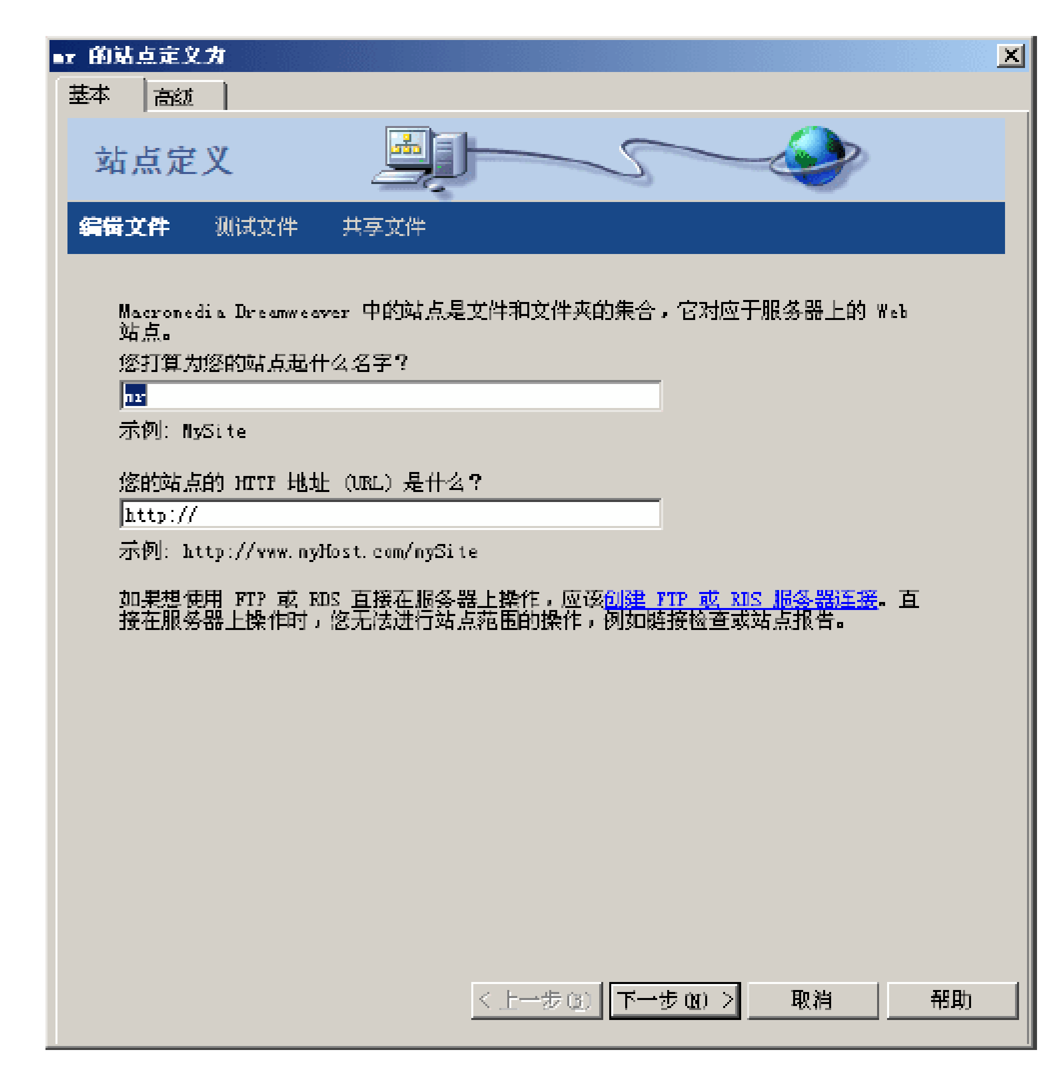


图 1.74　定义站点

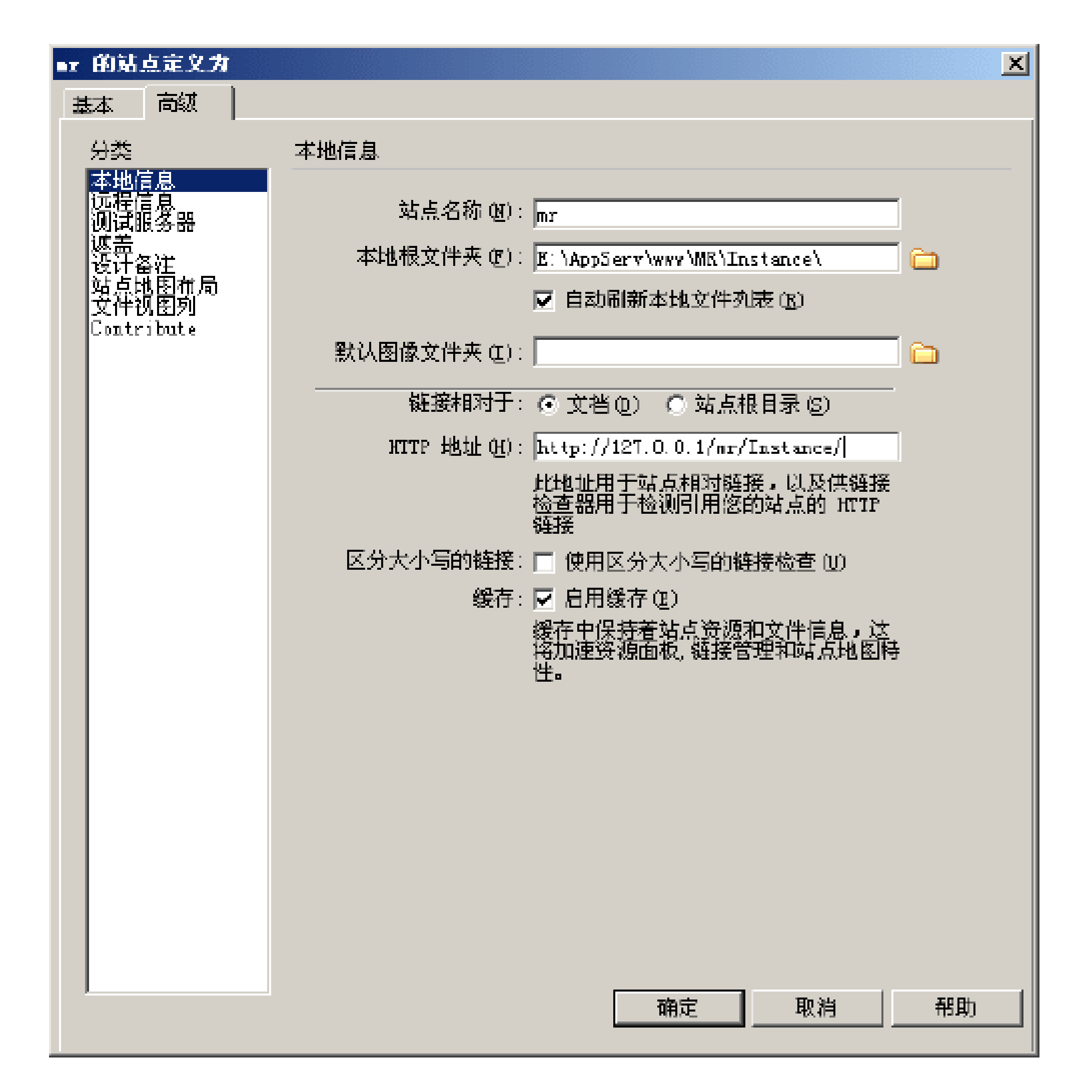


图 1.75　定义 mr 站点

（3）单击如图 1.75 所示对话框左侧的“测试服务器”，弹出如图 1.76 所示的测试服务器对话框，选择服务器模型：PHP MySQL，访问：本地/网络，测试服务器文件夹：E:\AppServ\www\MR\Instance\，URL 前缀：http://localhost/mr/Instance/，最后单击“确定”按钮。

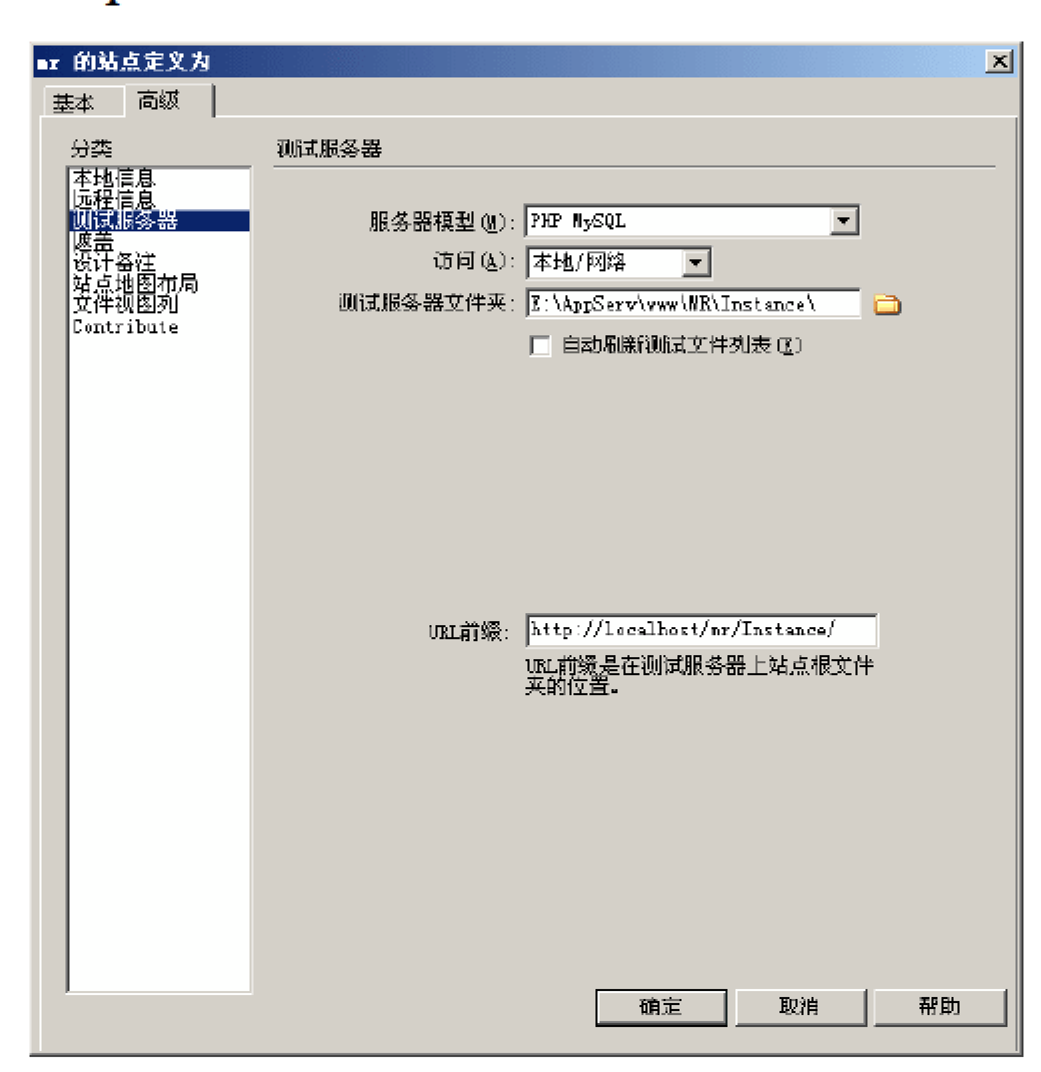


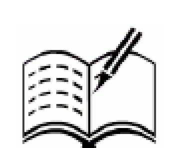
图 1.76　配置测试服务器

（4）mr 站点和测试服务器设置完毕，在 Dreamweaver 下直接使用快捷键 F12 即可浏览程序。

专家点评

在 Dreamweaver 中创建站点和配置测试服务器时，如果本地的 HTTP 地址与测试服务器中的 URL 前缀不统一，那么就无法通过 F12 键直接浏览程序。





Note

问题 19　如何使用 Dreamweaver 开发 PHP 程序?

问题阐述

Dreamweaver 是 Adobe 公司的 Web 站点和应用程序的专业开发工具，它将可视布局工具、应用程序开发功能和代码编辑组合在一起。其功能强大，使得各个层次的设计人员和开发人员都能够美化网站及创建应用程序。从基于 CSS 设计的领先支持手工编码，Dreamweaver 为专业人员提供了一个集成、高效的环境，这样开发人员就可以使用 Dreamweaver 及所选择的服务器来创建功能强大的 Internet 应用程序，从而使用户能连接到数据库、Web 服务和旧式系统。那么该如何使用 Dreamweaver 建立站点及开发 PHP 程序呢？

专家解答

利用 Dreamweaver 建立站点及开发 PHP 程序的具体步骤如下：

（1）在 Dreamweaver 下建立站点。首先应创建一个文件夹，作为站点的根目录。

（2）打开 Dreamweaver，选择“站点”→“新建站点”命令，如图 1.77 所示。

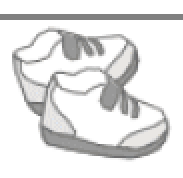


图 1.77　选择新建站点

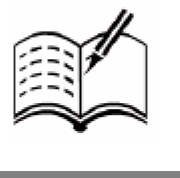
（3）打开“新建站点”窗口，如图 1.78 所示。在该窗口中选择“基本”选项卡，在“您打算为您的站点起什么名字”文本框中输入要创建的站点名称；在“您的站点的 HTTP 地址（URL）是什么？”文本框中输入运行时的 URL 地址，定位到创建的根目录下。

（4）选择图 1.78 中的“高级”选项卡，打开如图 1.79 所示的窗口，其中的“站点名称”文本框中的内容自动更改为在“基本”选项卡中为站点所取的名称“01”。在“本地根文件夹”文本框中输入或选择在步骤（1）中建立的网站根文件夹“F:\AppServ\www\mr\01\”。单击“确定”按钮返回“站点管理”窗口，再单击该窗口的“完成”按钮，成功创建站点。

（5）选择“文件”→“新建”命令，将弹出如图 1.80 所示的对话框。在该对话框的类别列表中选择“动态页”选项，然后在动态页列表中选择 PHP 选项，单击“创建”按钮，在 Dreamweaver 中创建 PHP 文件，在该文件中输入如下代码：



```
<?php
echo "通过Dreamweaver开发PHP程序";
?>
```



Note



图 1.78 “01 的站点定义为”窗口中的“基本”选项卡



图 1.79 “01 的站点定义为”窗口中的“高级”选项卡

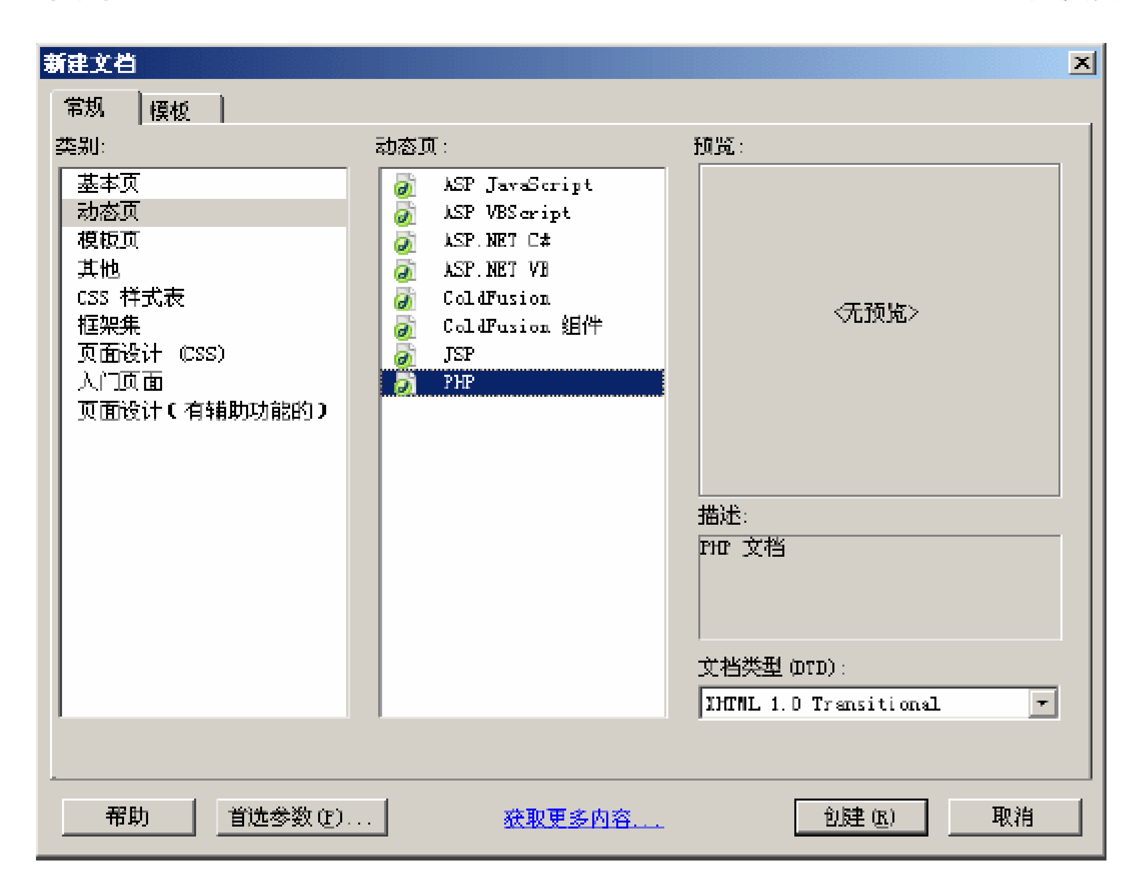


图 1.80 新建 PHP 文件

（6）将该文件命名为 index.php，保存在创建的站点的根目录下，如图 1.81 所示。

（7）打开浏览器，在地址栏中输入 http://192.168.1.59/mr/01/005/sl//index.php，如果浏览器中输出如图 1.82 所示的页面，则说明利用 Dreamweaver 开发 PHP 程序得以实现。

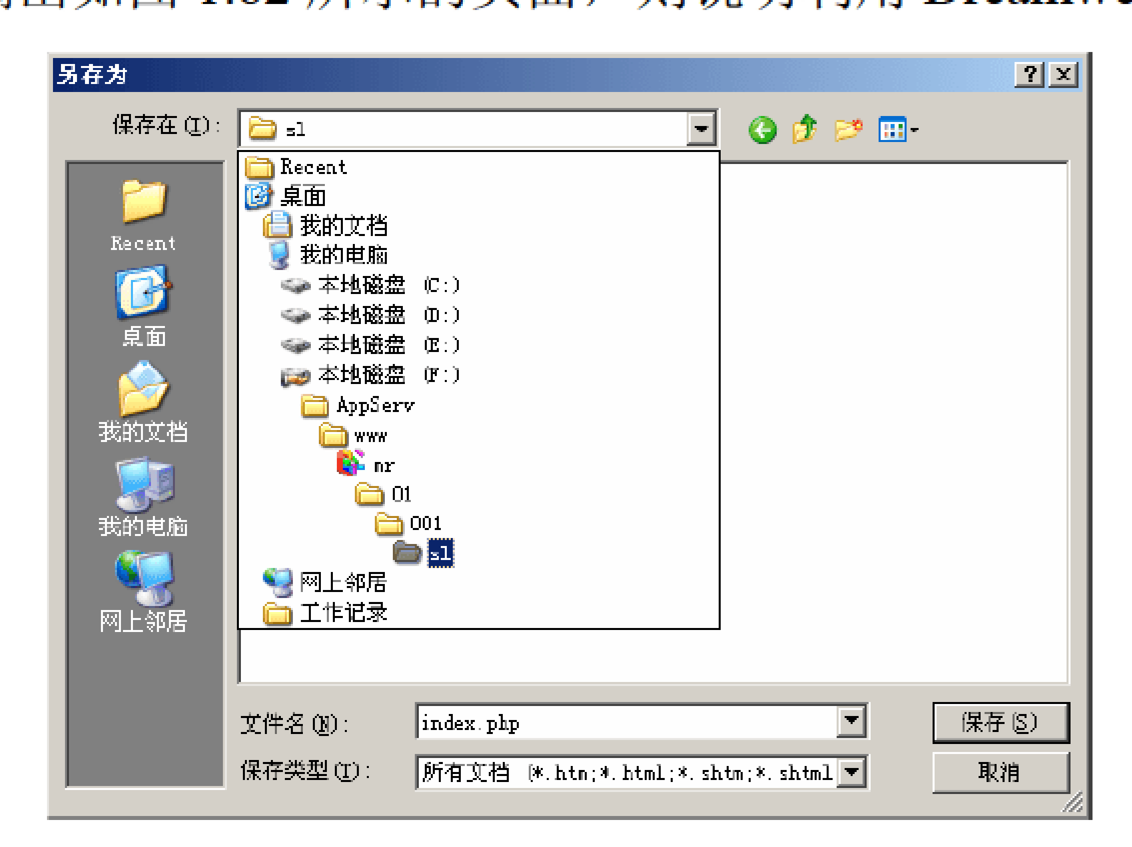


图 1.81 文件的保存

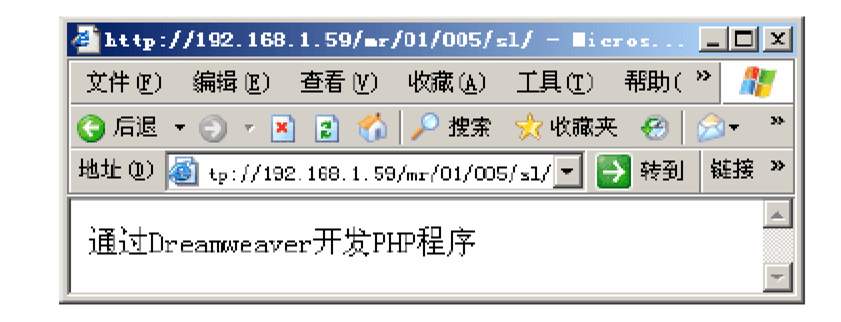


图 1.82 程序运行结果



专家点评

在通过 Dreamweaver 开发 PHP 程序的过程中，如果想要在 Dreamweaver 中使用快捷键 F12 来浏览程序，那么必须要在创建站点时对服务器进行配置，否则将不能使用快捷键来浏览程序。

问题 20　如何使用 Dreamweaver 识别.phtml 的模板文件？

问题阐述

如何使用 Dreamweaver 识别.phtml 格式的模板文件呢？

专家解答

解决方法如下：

（1）找到 Dreamweaver 安装目录下的 Configuration\DocumentTypes\MMDocumentTypes.xml 文件并将其打开。

（2）将 MMDocumentTypes.xml 文件中的如下代码：

```
<documenttype id="PHP_MySQL" servermodel="PHP MySQL" internaltype="Dynamic" winfileextension="php,php3,php4,php5" macfileextension="php,php3,php4,php5" file="Default.php" writebyteordermark="false">
```

改为：

```
<documenttype id="PHP_MySQL" servermodel="PHP MySQL" internaltype="Dynamic" winfileextension="php,php3,php4,php5,phtml" macfileextension="php,php3,php4,php5,phtml" file="Default.php" writebyteordermark="false">
```

（3）启动 Dreamweaver 可以发现，以.phtml 为扩展名的模板文件也可以在其可视编辑模式下进行编辑开发。

例如，使用 Dreamweaver 编辑 index.phtml 文件，如图 1.83 所示。

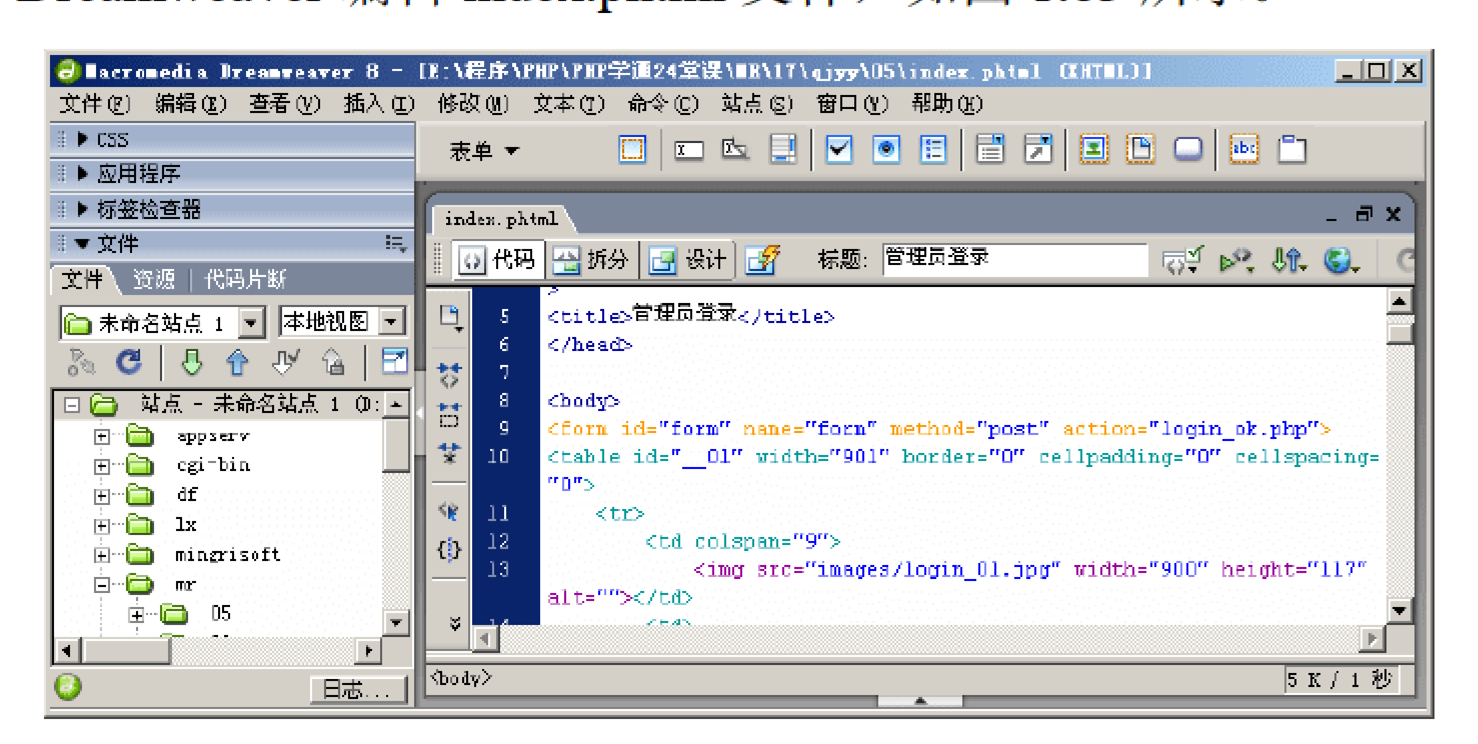
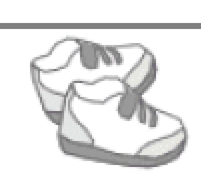


图 1.83　使用 Dreamweaver 编辑 index.phtml 文件

专家点评

PHTML（有时也称为 PHP）网页是一种包含 PHP（一种和 JavaScript 或 Microsoft VBScript 类似的语言）脚本的网页，和 ASP 一样，PHP 脚本也是镶嵌在网页的 HTML 代码之中的。在页面被发送给请求的用户之前，网页服务器调用 PHP 解释程序来解释和执行 PHP 脚本。含有



PHP 脚本的网页通常都以“.php”、“.php3”或“.phtml”作为后缀。和 ASP 一样，PHP 也可以被认为是一种“动态网页”。



问题 21　如何在 Windows 下安装和配置 phpMyAdmin?

问题阐述

如何在 Windows 下安装和配置 phpMyAdmin?

专家解答

1. 安装 phpMyAdmin 图形化管理工具

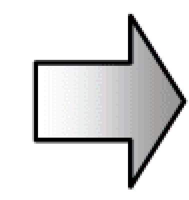
将下载的“phpMyAdmin-2.11.5-all-languages.rar”文件解压缩到 Apache 主目录下，解压后的名称是“phpMyAdmin-2.11.5-all-languages”，其中 2.11.5 是 phpMyAdmin 的版本号，为了便于使用可以将解压缩后的文件夹重新命名为 phpMyAdmin。

2. 配置 phpMyAdmin 图形化管理工具

将 phpMyAdmin 安装目录下的 libraries 目录下的 config.default.php 复制到 phpmyAdmin 根目录下，并改名为 config.inc.php。然后用 Dreamweaver 开发工具打开 config.inc.php 文件，将光标定位到指定的标识行位置。config.inc.php 文件初始值设置如图 1.84 所示，配置后的文件如图 1.85 所示。

```
39  $cfg['PmaAbsoluteUri'] = '';
64  $cfg['blowfish_secret'] = '';
151 $cfg['Servers'][$i]['auth_type'] ='config';
158 $cfg['Servers'][$i]['user'] = 'root';
165 $cfg['Servers'][$i]['password'] = '';
```

图 1.84　配置 config.inc.php 文件前



```
39  $cfg['PmaAbsoluteUri'] = 'http://localhost/phpmyadmin';
64  $cfg['blowfish_secret'] = 'cookie';
151 $cfg['Servers'][$i]['auth_type'] = 'http';
158 $cfg['Servers'][$i]['user'] = 'root';
165 $cfg['Servers'][$i]['password'] = 'root';
```

图 1.85　配置 config.inc.php 文件后

第 39 行用于设置 phpMyAdmin 网址。

第 64 行用于设置访问方式为 cookie，本机不需要设置，但是网络需要设置成 cookie。

第 151 行用于设置 phpMyAdmin 的登录方式。可以设置的登录方式有 3 种：config、http 和 cookie。其中 config 是默认的登录方式，该方式表示用户正确地输入用户名和密码后可直接登录；如果使用 http 方式登录，会弹出一个对话框，提示输入用户名和密码；如果使用 cookie 方式登录，在访问 phpMyAdmin 时会打开一个登录页面，提示用户输入用户名和密码，并将用户名和密码保存在系统的 cookie 中，下次访问时用户名和密码将会自动显示在文本框和密码输入框中，通常用于互联网。

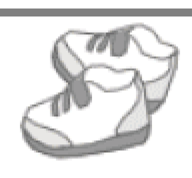
第 158 行用于设置 MySQL 数据库的用户名。

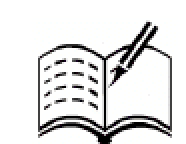
第 165 行用于设置 MySQL 数据库的密码。

这里设置的用户名和密码是数据库中已经存在的用户名和密码，如果设置的用户名和密码在数据库中不存在，那么将不可以使用。

说明：

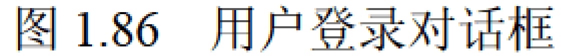
如果读者未使用 Dreamweaver 工具打开 config.inc.php 文件，那么对应的行号会有所差异，读者可以搜索相关的关键字，如，“['auth_type']”进行搜索定位，然后更改对应的配置文件。





设置完毕后，保存 config.php 文件，在浏览器中输入“http://localhost/phpMyAdmin”，将弹出用户登录对话框，如图 1.86 所示。




图 1.86　用户登录对话框


此时输入正确的用户名和密码（笔者在安装 MySQL 时，设置 MySQL 数据库的用户名是“root”，密码是“root”），即可进入 phpMyAdmin 的主页面。

问题 22　如何在 Linux 下安装和配置 phpMyAdmin？

问题阐述

如何在 Linux 下安装和配置 phpMyAdmin？

专家解答

1. 安装 phpMyAdmin 图形化管理工具

在 Linux 操作系统下安装 phpMyAdmin，将下载的软件包“phpMyAdmin-2.11.5-all -languages.rar”解压缩到配置环境的根目录下，为了便于使用，可以将解压缩后的文件夹重新命名为 phpMyAdmin。

命令模式下的操作如下：

```
# mv phpMyAdmin-2.11.5-all-languages.rar/var/ww/html
# tar zvxf phpMyAdmin-2.11.5-all-languages.rar
# mv phpMyAdmin-2.11.5-all-languages    phpMyAdmin
```

通过上述 3 个命令将下载的软件包解压缩到/var/www/html/目录下，从而完成 phpMyAdmin 图形化管理工具的安装。

2. 配置 phpMyAdmin 图形化管理工具

配置在 phpMyAdmin 目录下的 config.inc.php 文件下进行，更改如下参数的配置。

```
//将参数 PmaAbsoluteUri 的值设定为 phpMyAdmin 所处的位置:
$cfg['PmaAbsoluteUri'] = 'http://localhost/phpMyAdmin';
$cfg['blowfish_secret']='cookie';              //设定访问方式为 Cookie
$cfg['Servers'][$i]['host'] = 'localhost';     //设定 MySQL 所在的主机名或 IP 地址
$cfg['Servers'][$i]['auth_type'] = 'http';     //设定进入 phpMyAdmin 管理 MySQL 的方式
$cfg['Servers'][$i]['user'] = 'root';          //设定管理 MySQL 的账号
$cfg['Servers'][$i]['password'] = 'root';      //设定管理 MySQL 的密码
```

参数设置完成后保存该文件，重新启动 Apache 服务器。



在浏览器的地址栏中输入“http://localhost/phpMyAdmin/”即可打开 phpMyAdmin 图形化管理主页面进行数据库操作。



问题 23　如何应用 phpMyAdmin 修改数据库的用户名和密码？

问题阐述

修改 MySQL 数据库的用户名和密码的方法有很多，应用 phpMyAdmin 来修改是十分方便快捷的方法，那么如何操作呢？

专家解答

通过 phpMyAdmin 图形化管理工具来修改数据库用户名和密码的具体操作如下。

（1）单击 phpMyAdmin 主页面中的权限超链接，打开服务器用户操作页面，如图 1.87 所示。在该页面中，可以对指定用户的权限进行编辑，可以添加新用户和删除指定的用户。这里选择指定的用户，单击（编辑权限）超链接，对指定用户的权限进行设置，进入如图 1.88 所示的页面。

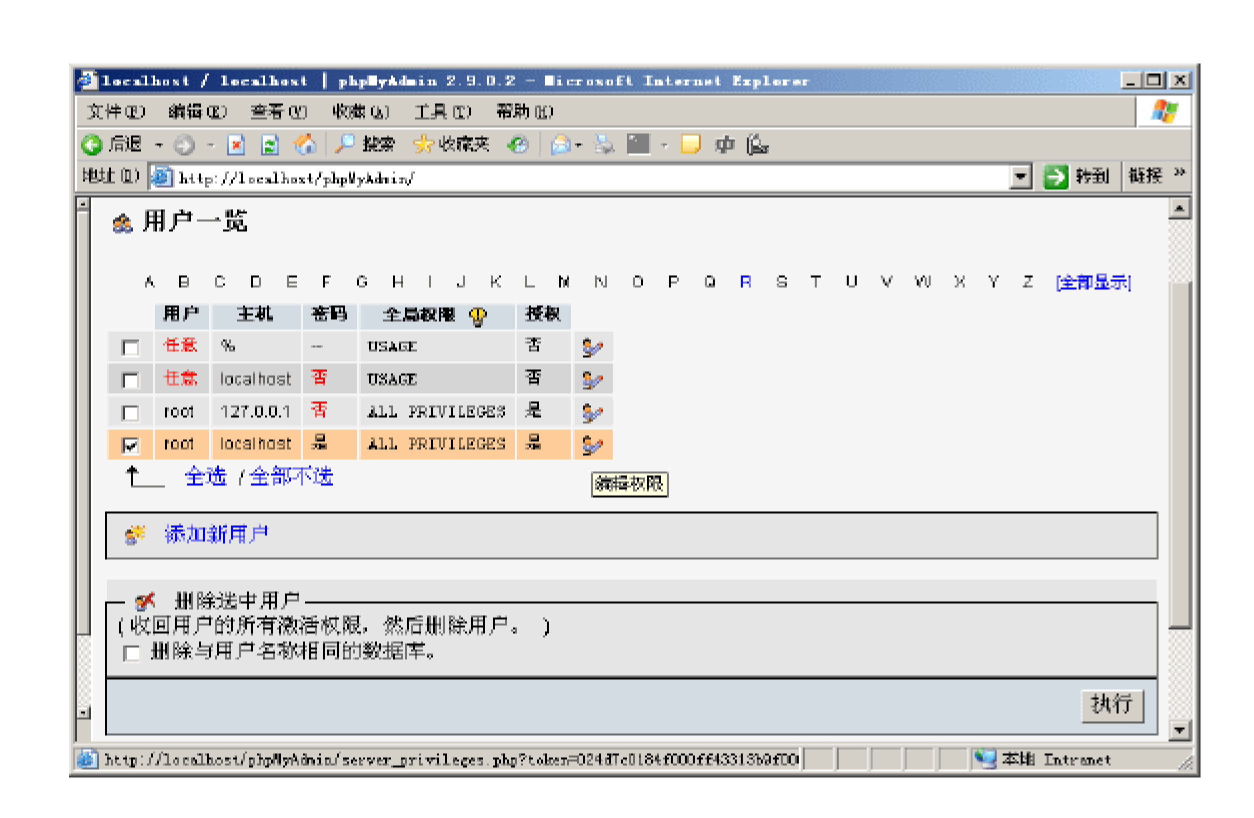


图 1.87　服务器用户一览表

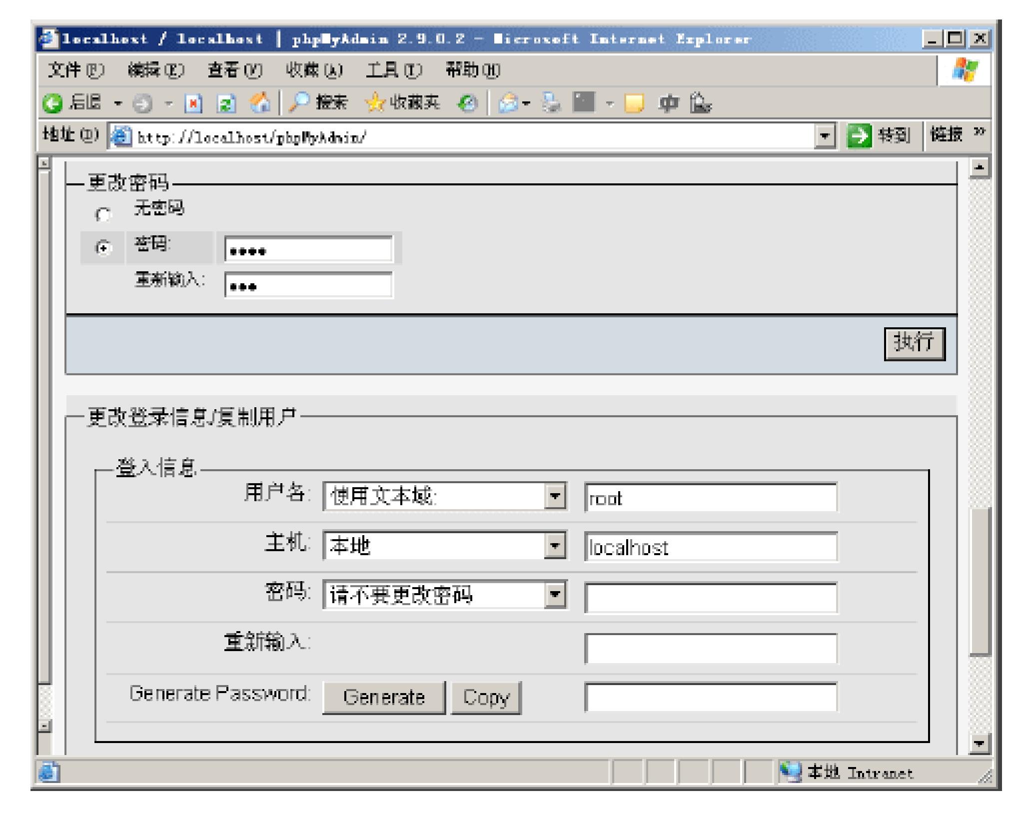


图 1.88　编辑用户权限

（2）在图 1.88 所示的页面中，可以设置用户的权限、修改密码、更改登录用户的信息和复制用户。在输入原密码和新密码之后，单击“执行”按钮即可完成对用户密码的修改操作，返回主页面，如图 1.89 所示。

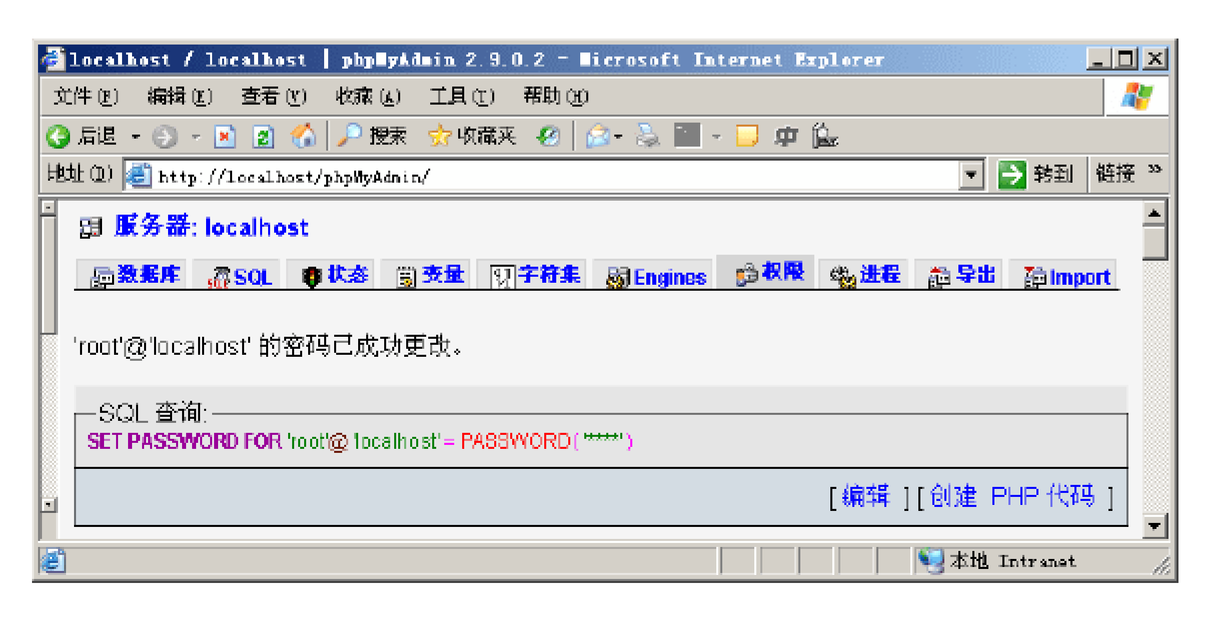
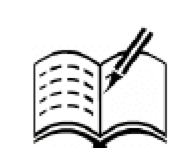


图 1.89　用户密码修改成功





问题 24　如何应用 phpMyAdmin 创建数据库与数据表？

问题阐述

数据库与数据表的创建可以通过命令符来实现，但是使用该方法对于初学者来说比较麻烦，而且因其需要手写的代码比较多，很容易出现错误，那么如何才能简单而直接地创建数据库与数据表呢？

专家解答

1. 创建数据库

在 phpMyAdmin 主页面的“创建一个新的数据库”文本框中输入数据库的名称“db_study”，然后在其下方的下拉列表框中选择所要使用的编码，一般选择“gb2312_Chinese_ci”简体中文编码格式，单击“创建”按钮，创建数据库，如图 1.90 所示。

数据库创建成功后，将显示如图 1.91 所示的页面。

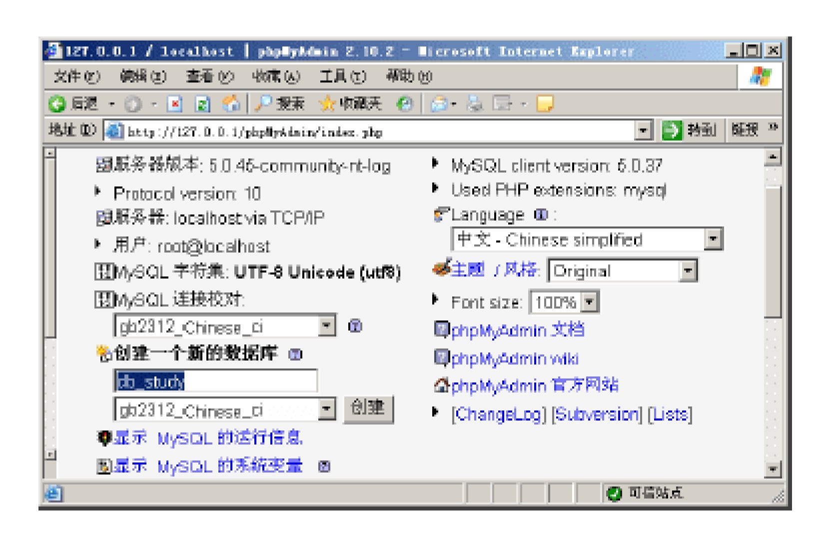


图 1.90　phpMyAdmin 管理主页面

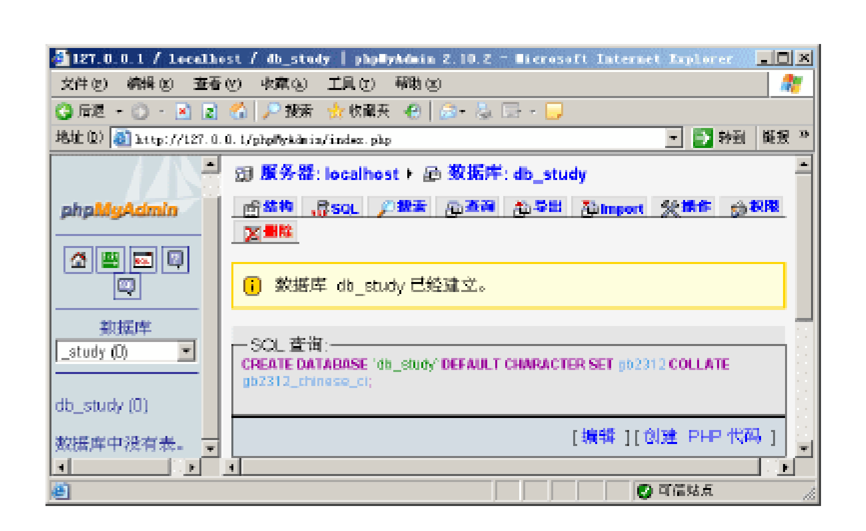


图 1.91　数据库创建成功

说明：

在图 1.91 所示页面的右侧区域中可以对该数据库进行相关操作，如结构、SQL、导出、搜索、查询、删除等，单击相应的超链接即可进入相应的操作页面。但是在创建的数据库还没有创建数据表的情况下，只能够执行结构、SQL、Import、操作、权限和删除 6 项操作，而搜索、查询和导出这 3 项操作不能执行，当光标指向这几个超链接时，将弹出不可用标记。

2. 创建数据表

数据库创建成功后，在该数据库下创建 tb_user 数据表。在 db_study 数据库右侧的操作页面中输入数据表的名称和字段数，然后单击“执行”按钮，即可创建数据表，如图 1.92 所示。

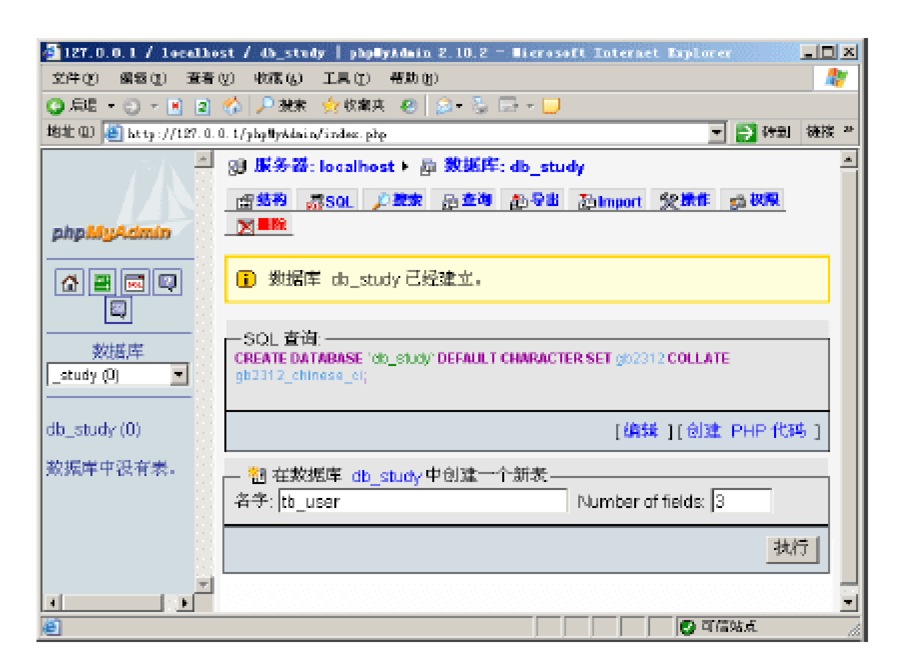
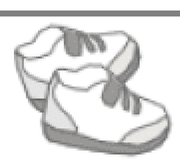


图 1.92　创建数据表



成功创建数据表 tb_admin，将显示数据表结构页面。在表单中对各个字段的详细信息进行录入，包括字段名、数据类型、长度/值、编码格式、是否为空、主键等，以完成对表结构的详细设置。当所有的信息都输入完成以后，单击“保存”按钮，创建数据表结构，如图 1.93 所示。



注意：

在定义数据表的格式时必须指定主键，其类型为 int。

数据表创建成功，效果如图 1.94 所示。

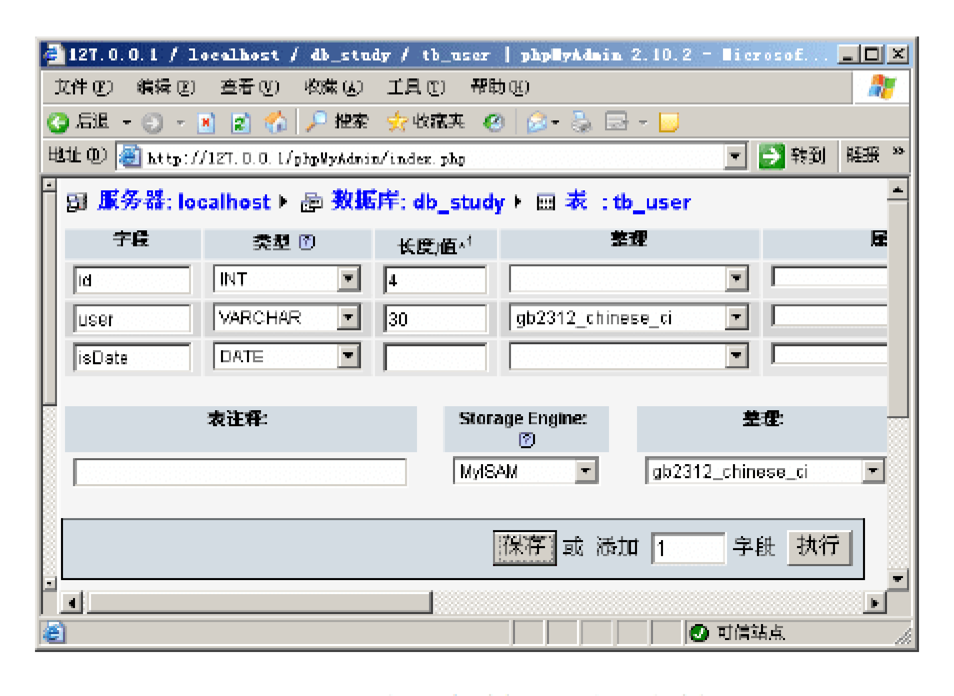


图 1.93　创建数据表结构

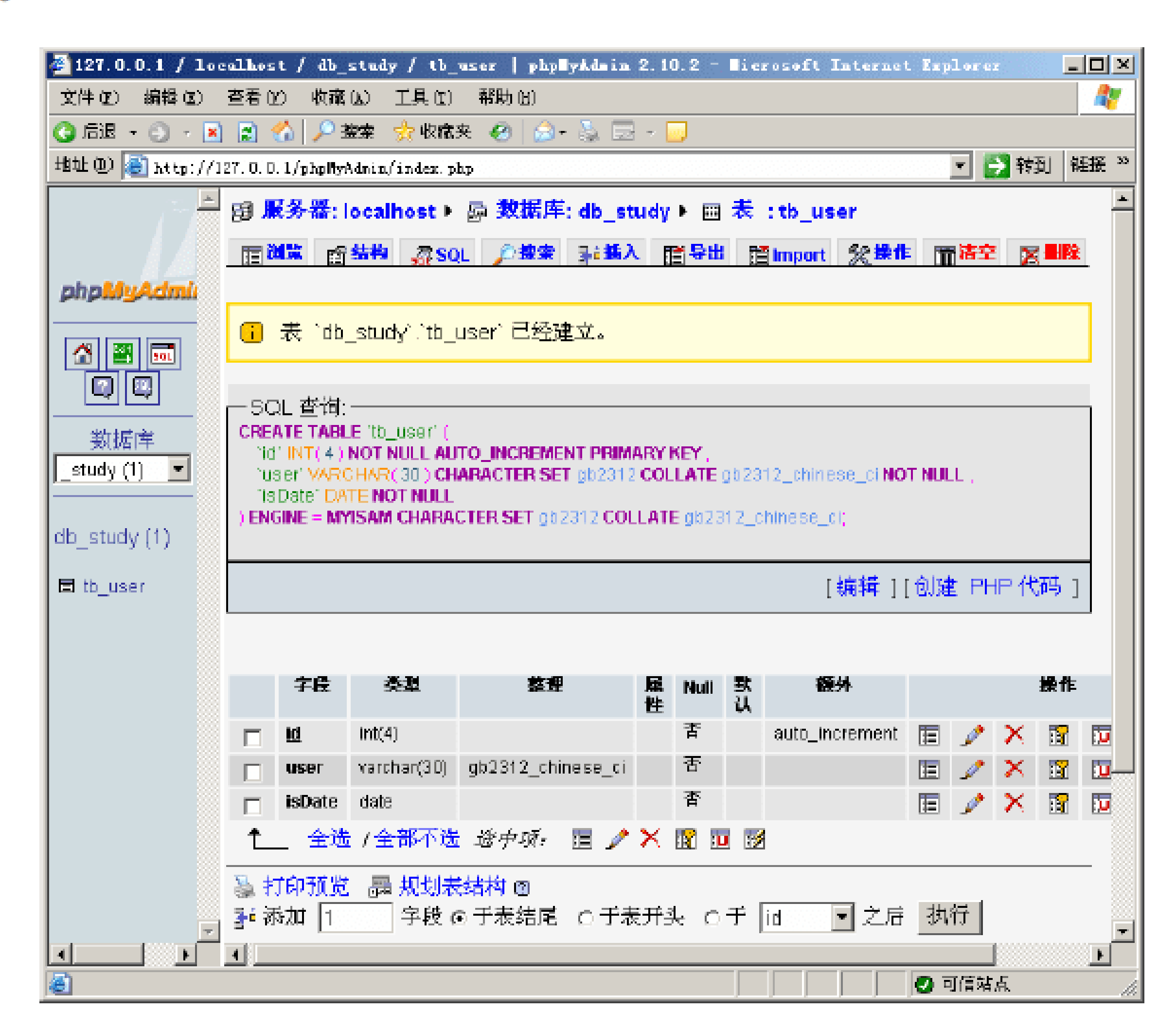


图 1.94　tb_user 数据表

问题 25　如何应用 phpMyAdmin 导入和导出 MySQL 数据库脚本？

问题阐述

phpMyAdmin 是一个使用 PHP 编写的、基于 Web 的 MySQL 管理工具，可以通过互联网控制和操作 MySQL。那么该如何应用 phpMyAdmin 导入和导出 MySQL 数据库脚本呢？

专家解答

导入和导出 MySQL 数据库脚本是互逆的两个操作。导入是执行扩展名为“.sql”的文件，将数据导入数据库中；导出是将数据表结构、表记录存储为“.sql”的脚本文件。通过导入和导出的操作实现数据库的备份和还原。

1. 导出 MySQL 数据库脚本

单击 phpMyAdmin 主页面中的超链接，打开导出编辑区，如图 1.95 所示。选择导出文件的格式，这里默认使用选项“SQL”，选中“另存为文件”复选框，单击“执行”按钮，弹出如图 1.96 所示的文件下载对话框，单击“保存”按钮，将脚本文件以“.sql”格式存储在指定位置。





Note

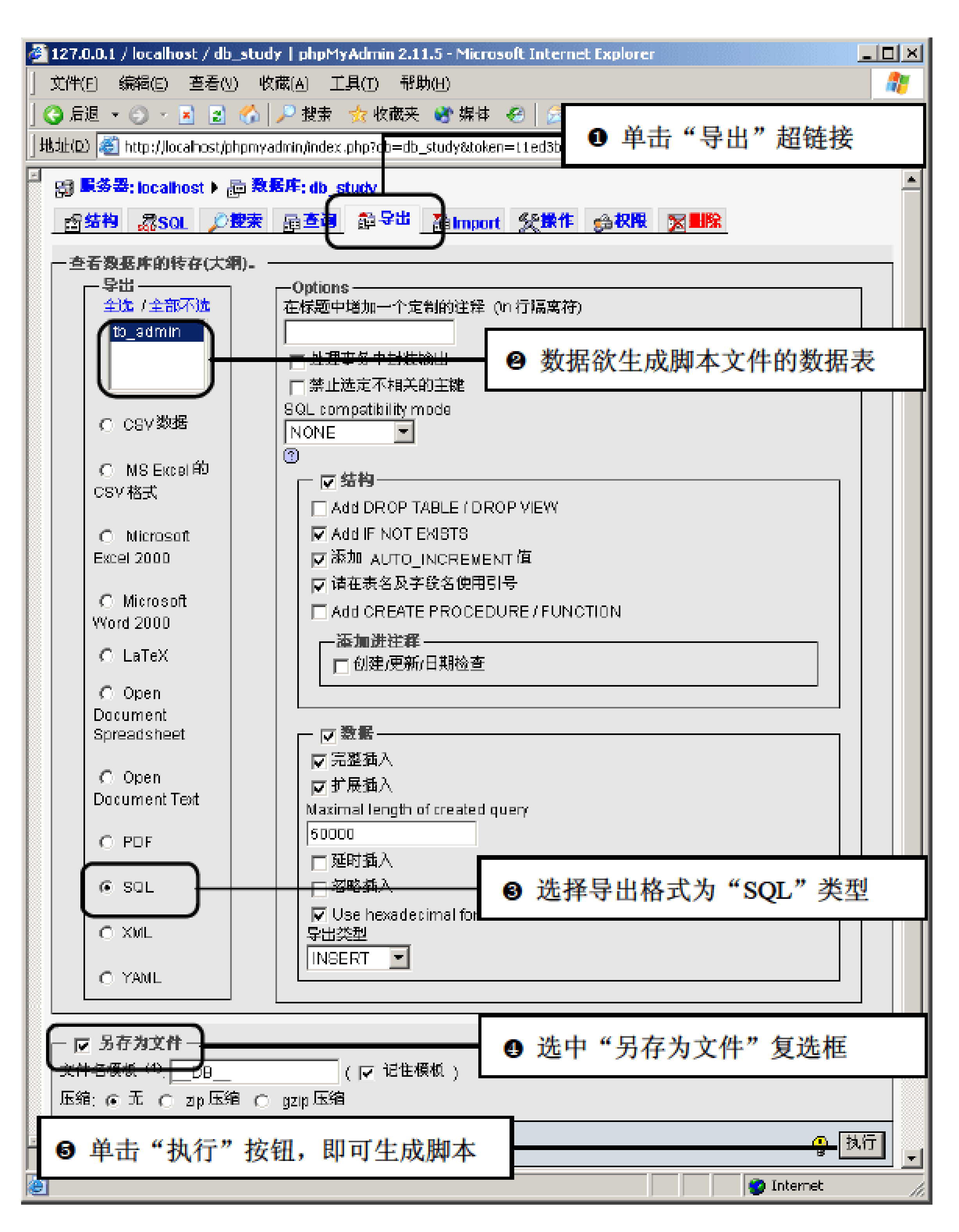


图 1.95　生成 MySQL 脚本文件设置页面

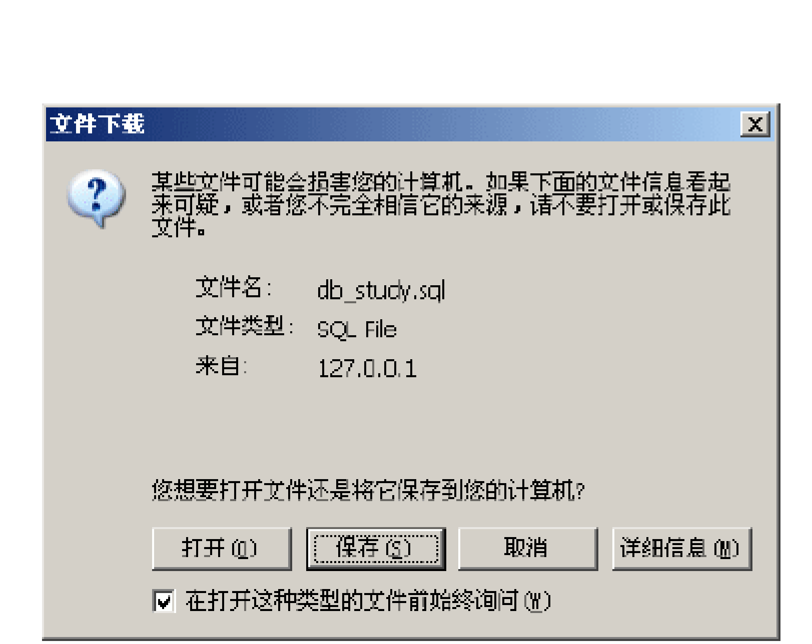


图 1.96　存储 MySQL 脚本对话框

2. 导入 MySQL 数据库脚本

单击 Import 超链接，进入执行 MySQL 数据库脚本页面，单击"浏览"按钮查找脚本文件（如 db_study.sql）所在位置，如图 1.97 所示，单击"执行"按钮，即可执行 MySQL 数据库脚本文件。

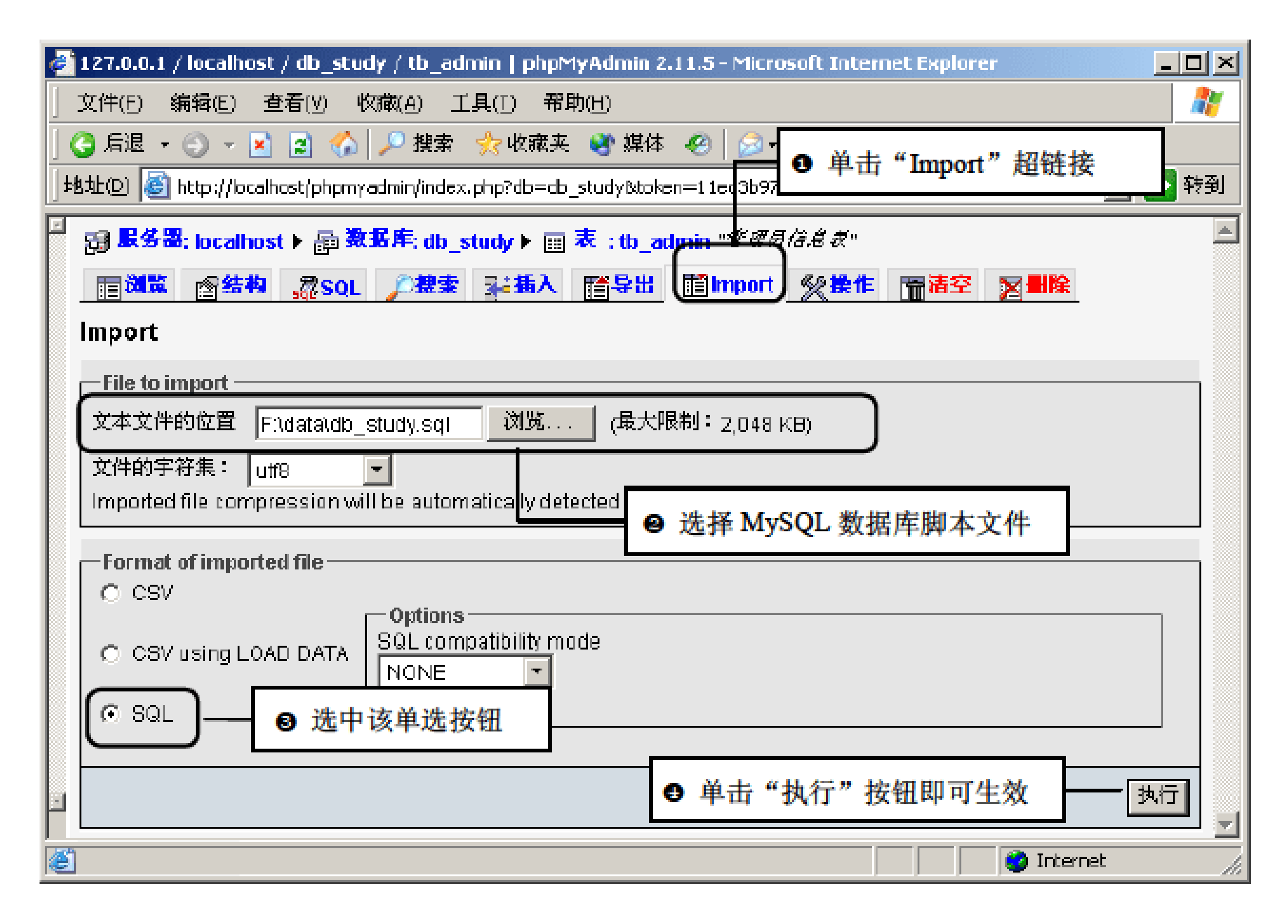
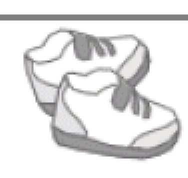


图 1.97　执行 MySQL 数据库脚本文件

专家点评

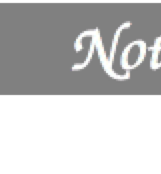
在执行导入操作之前，首先要检测数据库中是否存在与所导入数据库同名的数据库，如果不存在，则首先要在数据库中创建一个名称与数据文件中的数据库名相同的数据库，然后再执



行 MySQL 数据库脚本文件。另外，在当前数据库中，不能有与将要导入数据库中的数据表重名的数据表存在，如果存在则导入文件就会失败，并提示错误信息。

另外，读者也可通过单击 phpMyAdmin 图形化工具左侧区的按钮，在打开的对话框中单击“导入文件”超链接，然后选择脚本文件所在的位置，从而执行数据导入操作。





问题 26　如何打开 magic_quotes_gpc 来防止 SQL 注入?

问题阐述

SQL 注入是非常危险的事情，轻则网站后台被入侵，重则整个服务器沦陷，所以用户一定要小心。那么该如何防止 SQL 的注入呢？

专家解答

php.ini 中有一个设置：

```
magic_quotes_gpc = Off
```

该设置默认是关闭的，如果它打开后将自动把用户提交对 sql 的查询进行转换，例如把“'”转换为“\'”等，这对防止 sql 注入有重大作用。所以我们将 magic_quotes_gpc 设置为 On。设置代码如下：

```
magic_quotes_gpc = On
```

问题 27　如何对错误提示信息进行控制?

问题阐述

一般 php 在没有连接到数据库或其他情况下时会有错误提示，通常错误信息中会包含 php 脚本当前的路径信息或查询的 SQL 语句等信息，这类信息提供给黑客后是不安全的，所以一般服务器建议禁止错误提示。那么如何才能有效地控制错误信息提示呢？

专家解答

在 php.ini 文件中找到：

```
display_errors = On
```

将其改为：

```
display_errors = Off
```

即可将错误提示信息屏蔽掉。

如果一定要显示错误信息，可以设置显示错误的级别，如只显示警告以下的信息：

```
error_reporting = E_WARNING & E_ERROR
```

当然，错误提示关闭后，很难找到错误原因，所以，有需要时可以暂时开启，不需要时再将其关闭。



问题 28　如何记录错误日志？

问题阐述

为了方便查找服务器的运行原因，最好对错误日志进行记录，那么该如何记录日志呢？

专家解答

1. 设置错误日志存储位置

在 php.ini 文件中，关闭 display_errors 后把错误信息记录下来：

```
log_errors = On
```

同时也要设置错误日志存放的目录，建议和 Apache 的日志存在一起：

```
error_log = D:/usr/local/apache2/logs/php_error.log
```

> **注意：**
> Apache 用户必须具备错误日志文件的写权限，否则不能将错误信息添加到日志文件中。

2. 降低用户权限

在 MySQL 服务器中，默认使用的登录用户是本地系统用户，该用户的权限非常大，为了控制登录 MySQL 服务器用户的权限，可以通过创建新用户的方式来降低用户的权限。

其方法是应用 net user 命令创建新用户，并且将该用户从组中删除。其命令如下：

```
net user mr 123456/add
net localgroup users mr /del
```

创建名称为 mr 的用户，密码为 123456，并且将其从组中删除。

新的标准用户添加成功后，通过该新用户来登录 MySQL 服务器，这样就降低了用户的权限。其具体方法是：

在系统的“服务”中定位到 MySQL，右击 MySQL，在弹出的快捷菜单中选择“属性”命令，在弹出的对话框中选择“登录”选项卡，如图 1.98 所示。

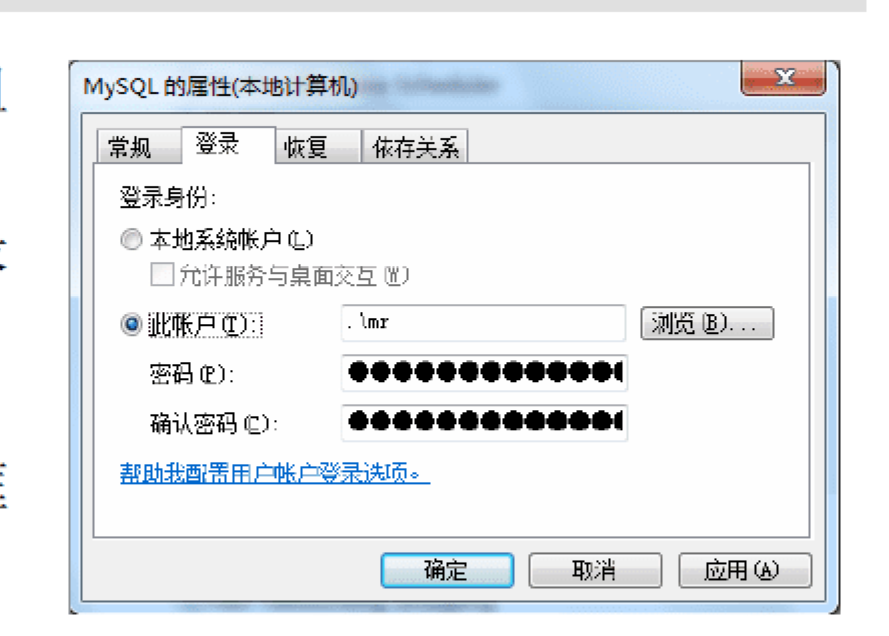


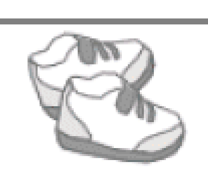
图 1.98　设置 MySQL 的登录用户

在“登录身份”选项组中选中“此账户”单选按钮，找到所创建的用户，并添加用户密码，单击“确定”按钮。重新启动 MySQL 服务，MySQL 即运行在低权限下。

在 Windows 操作系统下，Apache 默认运行的是 system 权限，可以应用上述的方法来降低 Apache 的权限。

专家点评

这里还可以通过设置各个文件夹的权限，来让 Apache 用户只能执行设定的事情，为每一个目录建立一个单独能读写的用户。这也是当前很多虚拟主机提供商的流行配置方法。



问题 29　如何解决使用 MySQL 函数导致 Apache(IIS)崩溃的问题?



Note

问题阐述

本问题中所说的崩溃是在 Windows 下配置 PHP 环境时发生的。具体的表现是 php 运行正常，但是只要一调用 MySQL 的相关函数，Apache 就会崩溃。

环境配置方式：

php 以模块的方式运行在 Apache 下。运行 phpinfo()函数，显示 mysql 库加载成功，但调用 MySQL 函数时则 Apache 崩溃，在 Windows 的错误日志中记录以下内容：

错误应用程序 httpd.exe，版本 2.2.11.0，错误模块 php5ts.dll，版本 5.2.10.10，错误地址 0x0000ac6a。

专家解答

在安装 MySQL 时，在配置设定时把 MySQL 的 Bin 目录放到了系统环境变量 Path 中。而 Bin 目录中正好有一个 libmysql.dll 的库。

而 PHP 在加载 MySQL 库时，会搜索系统环境变量 Path 的每一个目录寻找 libmysql.dll，由于 MySQL 的 bin 目录也在其中，那么 PHP 就先把 MySQL 的 libmysql.dll 加载了，所以就会由于库的兼容问题导致 MySQL 的函数无法运行。

解决方案：

将 MySQL 的 bin 目录从系统环境变量 Path 中移出，或者将 libmysql.dll 从 MySQL 的 bin 目录中移出，并且重新启动系统。

问题 30　HTTP Apache 错误代码都有哪些？分别代表什么含义？

问题阐述

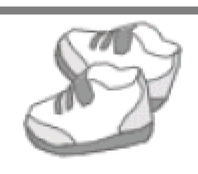
在运行 PHP 程序时，如果发生错误，都会返回一个错误代码，准确地解读这个错误代码的含义，对解决程序中的问题非常关键，有了它就可以准确地定位错误出现的位置。那么如何才能准确地解读错误代码呢？

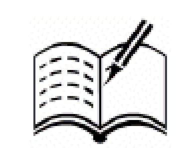
专家解答

为了帮助读者更好地解读 HTPP Apache 中的错误代码，特此对错误代码进行了汇总，让读者在学习过程中有一个参考。HTTP 状态码基本上可以分为 5 类：

1. 1xx 为消息类，该类状态代码用于表示服务器临时回应

100 Continue 表示初始的请求已经被服务器接受，浏览器应当继续发送请求的其余部分（HTTP 1.1）。





101 Switching Protocols 服务器将遵从客户的请求转换到另外一种协议（HTTP 1.1）。

2. 2xx 表示浏览器端请求被处理成功

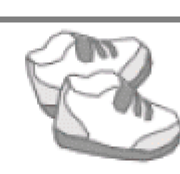
表 1.3　2xx 浏览器端请求处理代码说明

代　码	说　明
200 OK	一切正常
201 Created	服务器已经创建了文档，Location 头给出了它的 URL
202 Accepted	已经接受请求，但处理尚未完成
203 Non-Authoritative Information	文档已经正常地返回，但一些应答头可能不正确，因为使用的是文档的拷贝（HTTP 1.1新）
204 No Content	没有新文档，浏览器应继续显示原来的文档。这与表1.4中的304非常相似
205 Reset Content	没有新的内容，但浏览器应该重置它所显示的内容。用来强制浏览器清除表单输入内容（HTTP 1.1新）
206 Partial Content	客户发送了一个带有 Range 头的GET请求，服务器完成了它（HTTP 1.1新）。注意，通过 Range 可以实现断点续传

3. 3xx 重定向

表 1.4　3xx 重定向代码说明

代　码	说　明
300 Multiple Choices	客户请求的文档可以在多个位置找到，这些位置已经在返回的文档内列出。如果服务器要提出优先选择，则应该在Location应答头指明
301 Moved Permanently	客户请求的文档在其他地方，新的URL在Location头中给出，浏览器应该自动地访问新的URL
302 Found	类似于301，但新的URL应被视为临时性的替代，而不是永久性的。注意，在HTTP1.0中对应的状态信息是“Moved Temporatily”。出现该状态代码时，浏览器能够自动访问新的URL，因此它是一个很有用的状态代码。 注意：这个状态代码有时可以和301替换使用。例如，如果浏览器错误地请求http://host/~user（缺少了后面的斜杠），有的服务器返回301，有的则返回302。严格地说，我们只能假定只有当原来的请求是GET时浏览器才会自动重定向。可参见307
303 See Other	类似于301/302，不同之处在于，如果原来的请求是POST，Location头指定的重定向目标文档应该通过GET提取（HTTP 1.1新）
304 Not Modified	客户端有缓冲的文档并发出了一个条件性的请求（一般是提供If-Modified-Since头表示客户只想比指定日期更新的文档）。服务器告诉客户，原来缓冲的文档还可以继续使用
305 Use Proxy	客户请求的文档应该通过Location头所指明的代理服务器提取（HTTP 1.1新）
307 Temporary Redirect	和302(Found)相同。许多浏览器会错误地响应302应答进行重定向，即使原来的请求是POST，即使它实际上只能在POST请求的应答是303时才能重定向。由于这个原因，HTTP 1.1新增了307，以便更加清楚地区分几个状态代码：当出现303应答时，浏览器可以跟随重定向的GET和POST请求；如果是307应答，则浏览器只能跟随对GET请求的重定向（HTTP 1.1新）





4. 4xx 错误

表 1.5　4xx 错误代码说明

代　　码	说　　明
400 Bad Request	请求出现语法错误
401 Unauthorized	客户试图未经授权访问受密码保护的页面。应答中会包含一个WWW-Authenticate头，浏览器据此显示用户名字/密码对话框，然后在填写合适的Authorization头后再次发出请求
403 Forbidden	资源不可用。服务器理解客户的请求，但拒绝处理它。通常由于服务器上文件或目录的权限设置导致
404 Not Found	无法找到指定位置的资源。这也是一个常用的应答
405 Method Not Allowed	请求方法（GET、POST、HEAD、Delete、PUT、TRACE等）对指定的资源不适用（HTTP 1.1新）
406 Not Acceptable	指定的资源已经找到，但它的MIME类型和客户在Accpet头中所指定的不兼容（HTTP 1.1新）
407 Proxy Authentication Required	类似于401，表示客户必须先经过代理服务器的授权（HTTP 1.1新）
408 Request Timeout	在服务器许可的等待时间内，客户一直没有发出任何请求。客户可以在以后重复同一请求（HTTP 1.1新）
409 Conflict	通常和PUT请求有关。由于请求和资源的当前状态相冲突，因此请求不能成功。（HTTP 1.1新）
410 Gone	所请求的文档已经不再可用，而且服务器不知道应该重定向到哪一个地址。它和404的不同在于，返回407表示文档永久地离开了指定的位置，而404表示由于未知的原因文档不可用（HTTP 1.1新）
411 Length Required	服务器不能处理请求，除非客户发送一个Content-Length头（HTTP 1.1新）
412 Precondition Failed	请求头中指定的一些前提条件失败（HTTP 1.1新）
413 Request Entity Too Large	目标文档的大小超过服务器当前愿意处理的大小。如果服务器认为自己能够稍后再处理该请求，则应该提供一个Retry-After头（HTTP 1.1新）
414 Request URI Too Long	URI太长（HTTP 1.1新）
416 Requested Range Not Satisfiable	服务器不能满足客户在请求中指定的Range头（HTTP 1.1新）

5. 5xx 服务器错误

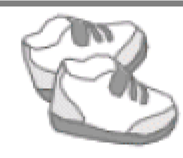
表 1.6　5xx 服务器错误代码说明

代　　码	说　　明
500 Internal Server Error	服务器遇到了未意料到的情况，不能完成客户的请求
501 Not Implemented	服务器不支持实现请求所需要的功能。例如，客户发出了一个服务器不支持的PUT请求
502 Bad Gateway	服务器作为网关或代理时，为了完成请求访问下一个服务器，但该服务器返回了非法的应答
503 Service Unavailable	服务器由于维护或负载过重未能应答。例如，Servlet可能在数据库连接池已满的情况下返回503。服务器返回503时可以提供一个Retry-After头
504 Gateway Timeout	由作为代理或网关的服务器使用，表示不能及时地从远程服务器获得应答（HTTP 1.1新）
505 HTTP Version Not Supported	服务器不支持请求中所指明的HTTP版本（HTTP 1.1新）

第 2 章

PHP 基础

- ⏭ 问题 31　提高编程效率的 24 招
- ⏭ 问题 32　为什么页面之间无法传递变量?
- ⏭ 问题 33　为什么 Win32 下 Apache2 用 get 方法传递中文参数会出错?
- ⏭ 问题 34　如何从文本中移除空白内容?
- ⏭ 问题 35　如何获取上传文件的后缀?
- ⏭ 问题 36　如何实现货币数据的格式化输出?
- ⏭ 问题 37　如何解决用 substr()函数对中文字符串截取时乱码?
- ⏭ 问题 38　如何获取多个数组元素的键名?
- ⏭ 问题 39　为什么$array[str]错了？
- ⏭ 问题 40　如何设置 PHP 的错误级别?
- ⏭ 问题 41　PHP 有哪几种注释形式?
- ⏭ 问题 42　如何使用 empty()函数判断文本框是否为空?
- ⏭ 问题 43　如何使用 strlen()函数计算文本长度?
- ⏭ 问题 44　如何使用 copy()函数对文件进行复制?
- ⏭ 问题 45　如何在浏览器中直接输出图像？（fgetc()函数）
- ⏭ 问题 46　如何关闭危险函数?
- ⏭ 问题 47　如何将数组中的元素合成字符串？
- ⏭ 问题 48　如何解决使用 empty 检查函数返回结果时报错?
- ⏭ 问题 49　如何使用 mb_substr 函数截取中文字符串?
- ⏭ 问题 50 如何解决 PHP 无法执行 system()函数的问题？
- ⏭ 问题 51　如何对查询关键字进行描红?
- ⏭ 问题 52　如何对字符串与 HTML 进行转换?
- ⏭ 问题 53　如何屏蔽违禁词语?
- ⏭ 问题 54　如何对字符串进行分割及连接?
- ⏭ 问题 55　如何将 IP 地址转换成对应的数字?
- ⏭ 问题 56　如何创建新型字符串?
- ⏭ 问题 57　如何对字符串进行加密解密?
- ⏭ 问题 58　如何比较字符串的大小?
- ⏭ 问题 59　如何获取上传文件的扩展名?
- ⏭ 问题 60　如何控制页面中输出字符串的长度?
- ⏭ 问题 61　如何定义上传文件的名称?
- ⏭ 问题 62　如何通过字符串截取控制图像验证码的输出?
- ⏭ 问题 63　如何打乱字符串中字符的顺序?
- ⏭ 问题 64　如何对字符串进行大小写转换?
- ⏭ 问题 65　函数 strstr()、stristr()、strpos()、strrchr()的区别是什么?
- ⏭ 问题 66　if 和 switch 语句有何区别?
- ⏭ 问题 67　如何防止页面出现中文乱码?

问题 31　提高编程效率的 24 招



Note

问题阐述

在开发 PHP 项目的过程中，要尽量采用标准的格式书写 PHP 代码，这不仅是作为一个程序员的良好习惯，同时还能够提高编程效率。那么在编程过程中，都有哪些是需要注意的呢？

专家解答

这里对编程中需要注意的一些规范、技巧进行了总结，供读者参考：

（1）用单引号代替双引号来包含字符串，这样做会更快一些。因为 PHP 会在双引号包围的字符串中搜寻变量，单引号则不会。注意：只有 echo 能这么做，它是一种可以把多个字符串当作参数的“函数”（说明：echo 是语言结构，不是真正的函数，故把函数加上了双引号）。

（2）$row['id']的速度要比$row[id]快很多。

（3）echo 比 print 快，并且使用 echo 的多重参数（说明：指用逗号而不是句号）代替字符串连接，比如 echo $str1,$str2。

（4）foreach 效率更高，尽量用 foreach 代替 while 和 for 循环。

（5）在执行 for 循环之前确定最大循环数，不要每循环一次都计算最大值，最好运用 foreach 代替。

（6）注销那些不用的变量尤其是大数组，以便释放内存。

（7）尽量不使用_get、_set、_autoload。

（8）使用 include()包含文件时，尽量使用绝对路径，因为它省去了 PHP 去 include_path 中查找文件的时间，解析操作系统路径所需的时间会更少。

（9）如果想获取脚本开始执行（说明：即服务器端收到客户端请求）的时间，使用$_SERVER['REQUEST_TIME']要优于 time()。

（10）在项目中尽量应用 PHP 内置函数实现功能，同时尽量使用函数代替正则表达式完成相同功能。

（11）str_replace 函数比 preg_replace 函数快，但 strtr 函数的效率比 str_replace 函数还要高。

（12）使用选择分支语句(switch)优于使用多个 if，else if 语句。

（13）不提倡使用@屏蔽错误消息，它的效率非常低。

（14）开启 Apache 的 mod_deflate 模块，可以提高网页的浏览速度。

（15）在连接数据库时，尽量使用短连接，要及时关闭使用完毕的数据库连接。

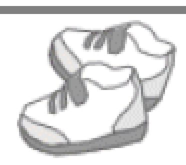
（16）在方法中递增局部变量，速度是最快的。几乎与在函数中调用局部变量的速度相当。

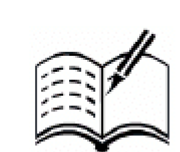
（17）派生类中的方法运行起来要快于在基类中定义的同样的方法。如果能将类的方法定义成 static，就尽量定义成 static，它的速度会提升很多。

（18）尽量少进行文件操作，虽然 PHP 的文件操作效率也不低。

（19）Apache 解析一个 PHP 脚本的时间要比解析一个静态 HTML 页面慢很多。尽量多用静态 HTML 页面，少用 PHP 脚本。

（20）除非脚本可以缓存，否则每次调用时都会重新编译一次。引入一套 PHP 缓存机制通常可以提升 25%~100%的性能，以免除编译开销。





（21）当操作字符串并需要检验其长度是否满足某种要求时，第一想法是使用 strlen()函数。此函数执行起来速度快，因为它不做任何计算，只返回在 zval 结构（C 的内置数据结构，用于存储 PHP 变量）中存储的已知字符串长度。但是，由于 strlen()是函数，多多少少会有些慢，因为函数调用会经过诸多步骤。在某些情况下，可以使用 isset()加速执行代码。例如：

```
<?php
if (strlen($foo) < 5) { echo "Foo is too short" $$ }
if (!isset($foo{5})) { echo "Foo is too short" $$ }
?>
```

调用 isset()恰巧比 strlen()快，因为与 strlen()不同的是，isset()作为一种语言结构，意味着它的执行不需要函数查找和字母小写化。也就是说，实际上在检验字符串长度的顶层代码中没有花太多开销。

（22）当执行变量$i 的递增或递减时，$i++会比++$i 慢一些（仅限 PHP 语言）。

++$i 更快是因为它只需要 3 条指令（opcodes），而$i++需要 4 条指令。后置递增实际上会产生一个临时变量，这个临时变量随后被递增，而前置递增直接在原值上递增。

（23）并不是事必面向对象（OOP），面向对象往往开销很大，每个方法和对象调用都会消耗很多内存。

（24）如果可以通过 file_get_contents 替代 file、fopen、feof、fgets 等系列方法的情况下，尽量用 file_get_contents，因为它的效率要高很多。但是，要注意 file_get_contents 在打开一个 URL 文件时的 PHP 版本问题。

问题 32　为什么页面之间无法传递变量?

问题阐述

为什么在表单中提交的数据，在其他的页面中无法获取？

专家解答

get、post、session 在最新的 PHP 版本中自动全局变量是关闭的(在 php.ini 中，register_globals 的值为 Off)，如果要从上一页面获取提交的变量，则必须使用$_GET['']、$_POST['']或$_SESSION['']。

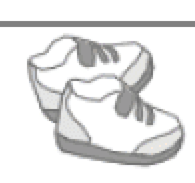
专家点评

从 PHP4.2.0 开始 register_globals 的默认值为 Off，这样一来，以前的很多可以直接使用的变量，如$PHP_SELF 或者设定的 SESSION 变量都不能用“$变量名”的形式访问了，这可能会给用户带来很多不便，但却有助于安全性的提高。访问这些变量，就需要使用 PHP 全局变量。

1. $_POST[]全局变量

使用 PHP 的$_POST[]预定义变量可以获取表单元素的值，格式为：

```
$_POST['name']
```





2. $_GET[]全局变量

使用 PHP 的$_GET[]预定义变量可以获取通过 GET 方法传过来的值，格式为：

```
$_GET['name']
```

3. $_SESSION[]变量

使用$_SESSION[]变量可以获取表单元素的值，格式为：

```
$_SESSION['name']
```

使用$_SESSION[]传参的方法获取的变量值，保存之后任何页面都可以使用，但这种方法很耗费系统资源，建议读者慎重使用。

注意：

PHP 可以应用$_POST[]或$_GET[]全局变量来获取表单元素的值。但值得注意的是，获取的表单元素名称区别字母大小写。如果读者在编写 Web 程序时疏忽了字母大小写，那么在程序运行时将获取不到表单元素的值或弹出错误提示信息。

问题 33　为什么 Win32 下 Apache2 用 get 方法传递中文参数会出错？

问题阐述

Win32 下 Apache2 用 get 方法传递中文参数会出错，如 test.php?a=你好&b=你也好，传递参数导致一个内部错误。

专家解答

要想解决这个问题，可以使用 urlencode()函数进行转换，如"test.php?a=".urlencode(你好)."&b=".urlencode(你也好)

专家点评

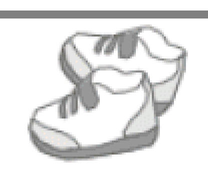
urlencode()函数主要是用于返回字符串类型，该函数会将传入的字符串进行编码 urlencode()函数与 rawencode()函数不同的是，urlencode()函数会将“空格”转换成“+”；而 rawencode()函数则会将“空格”转换成“%20”。urlencode()函数的其他字符串转换方式与 rawencode()函数的字符串转换方式相同。

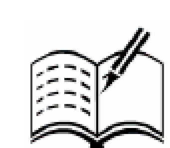
urlencode()函数的语法如下：

```
string urlencode (string str)
```

注意：

在使用 get 方法传递网址时，如果不使用编码方式则可能会出现获取的数据信息不完整的问题，因此，使用 urlencode()函数进行编码就会避免该问题的发生，而且又能保证传递数据的安全性，可谓是一举两得。





问题 34　如何从文本中移除空白内容？

问题阐述

在处理提交的内容时，经常需要考虑其中的空白内容。如果提交数据中存在空白，则会给应用程序造成问题。例如，错误地包含空白内容会导致存储的用户名或电子邮件地址不正确，因此，如果能够移除提交的表单值中的空白将会非常有用。

专家解答

在 PHP 中提供了 trim()函数去除字符串左右两边的空格和特殊字符；提供了 ltrim()函数去除字符串左边的空格和特殊字符；提供了 rtrim()函数去除字符串中右边的空格和特殊字符。

1. trim()函数

trim()函数用于去除字符串开始位置以及结束位置的空格，并返回去掉空格后的字符串。其语法如下：

```
string trim(string str [,string charlist]);
```

trim()函数的参数 str 是要操作的字符串对象，参数 charlist 为可选参数，指定需要从指定的字符串中删除哪些字符，如果不设置该参数，则所有的可选字符都将被删除。trim()函数的参数 charlist 的可选值如表 2.1 所示。

表 2.1　trim()函数的参数 charlist 的可选值

参　数　值	说　　明
\0	NULL，空值
\t	tab，制表符
\n	换行符
\x0B	垂直制表符
\r	回车符
" "	空格

说明：

除了以上默认的过滤字符列表外，也可以在 charlist 参数中提供要过滤的特殊字符。

2. ltrim()函数

ltrim()函数用于去除字符串左边的空格或指定字符串。其语法如下：

```
string ltrim(string str [,string charlist]);
```

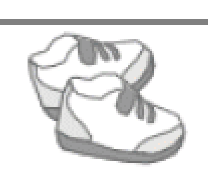
3. rtrim()函数

rtrim()函数用于去除字符串右边的空格。其语法如下：

```
String rtrim(string str [,string charlist]);
```

专家点评

在使用 ltrim()函数和 rtrim()函数时，默认情况下是只去除左边或右边的空格，想要去除特



Note

殊字符需要在函数中注明。

问题 35　如何获取上传文件的后缀?

问题阐述

文件上传功能几乎是所有网站所必备的，然而该功能对于服务器来说具有很大风险，因此应该对文件的大小和类型进行限制。可以通过获取上传文件的后缀来限制上传文件的类型。

专家解答

获取上传文件的后缀可以通过 strrev()函数对上传文件的名称进行反转输出，并通过 explode()函数以“.”为分隔符，对文件名称进行分割，然后再次应用 strrev()函数对数组中的第一个元素值进行返回输出，即获取到的就是上传文件的后缀。

strrev()函数实现 PHP 中字符串的逆序输出。其语法如下：

```
string strrev ( string string );
```

参数 string 指定需要反转的字符串。

例如，创建 PHP 脚本文件，当单击“上传”按钮时，首先，利用 post 方法接收文本框中的信息。其次，利用反转函数将文本框信息数据进行反转，并利用字符串拆分函数以“.”进行分割。然后，将分割结果保存在数组之中。最后，将指定数据再次反转取得并输出上传文件的后缀名称。其代码如下：

```
<?php
    if($_POST['sub']){                          //获取表单提交的数据
        $a = strrev($_POST['text']);            //字符串反转
        $b = explode(".",$a);                   //字符串分隔
        $c = strrev($b[0]);                     //获取文件后缀
        echo "文件后缀为:.$c";
    }
?>
```

专家点评

利用 strrev()函数检索字符串相对较麻烦，可以使用正则表达式函数 preg_match()来完成该功能。

问题 36　如何实现货币数据的格式化输出?

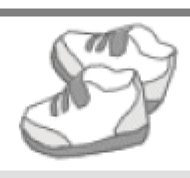
问题阐述

在开发电子网站时，在前台页面上输出货币数据显示的是整数数字，不能以货币的格式输出，那么如何才能让它以货币的格式输出呢？

专家解答

金额的格式化输出通过函数 number_format()实现。其语法说明如下：

```
string number_format ( float number , int decimals)
```



```
string number_format ( float number, int decimals, string dec_point, string thousands_sep)
```

number_format()函数返回参数 number 格式化后的字符串，该函数可以有一个、两个或 4 个参数，但不能有 3 个参数。如果只有一个参数 number，则 number 格式化后会舍去小数点后的值，且每一千就会以逗号（,）来隔开；如果有两个参数，则 number 格式化后会到小数点第 decimals 位，且每一千就会以逗号来隔开；如果有 4 个参数，则 number 格式化后会到小数点第 decimals 位，dec_point 用来替代小数点（.），thousands_sep 用来替代每一千隔开的逗号（,）。



问题 37 如何解决用 substr()函数对中文字符串截取时的乱码？

问题阐述

使用 substr()函数是按字节截取字符串的，在截取中文字符串时由于一个汉字由两个字符组成，所以如果只截取一个字符就会出现乱码。

专家解答

通过 substr()函数对中文字符串进行截取会出现乱码的问题，那么我们可以在 substr()函数的基础上封装一个自定义函数，在进行截取之前判断其是中文字符串还是英文字符串，如果是中文字符串则每次截取两个字节，如果是英文字符串则每次截取一个字节，这样就不会出现乱码的问题了。自定义函数的语法如下：

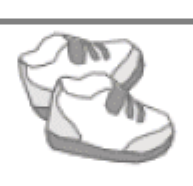
```
function msubstr($str, $start, $len) { //$str 指的是字符串，$start 指的是字符串的起始位置，$len 指
                                         的是长度
    $strlen = $start + $len;            //用$strlen 存储字符串的总长度（从字符串的起始位置到字符
                                         串的总长度）
    for($i = 0; $i < $strlen; $i ++) { //通过 for 循环语句，循环读取字符串
        if (ord (substr ($str, $i, 1) ) > 0xa0) {//如果字符串中首个字节的 ASCII 序数值大于 0xa0，
                                                  则表示为汉字
            $tmpstr .= substr ( $str, $i, 2 );//每次取出两位字符赋给变量$tmpstr，即等于一个汉字
                                              $i ++; //变量自加 1
        } else {                        //如果不是汉字，则每次取出一位字符赋给变量$tmpstr
            $tmpstr .= substr ( $str, $i, 1 );
        }
    }
    return $tmpstr;                     //输出字符串
}
```

参数$str 是指定被截取的字符串；参数$strart 是截取的开始位置；参数$len 是截取的长度。返回值$tmpstr 是截取后的字符串。

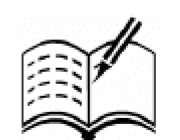
专家点评

对中文字符串的截取虽说是通过自定义函数来完成的，但是其根本还是应用 substr 函数，只是在进行字符串截取时，对字符串的类型进行了判断。

对截取字符串中首个字节的 ASCII 序数值进行判断，如果 ASCII 序数值大于 0xa0，则表



示为汉字，那么在应用 substr 函数进行截取时，就以两个字节为单位进行截取；如果 ASCII 序数值小于 0xa0，则表示为英文字符串，那么在应用 substr 函数进行截取时，就以一个字节为单位进行截取。这样将中文字符串和英文字符串分隔进行截取就避免了出现乱码的问题。



Note

问题 38　如何获取多个数组元素的键名？

问题阐述

如何获取多个数组元素的键名？

专家解答

在使用 array_search()函数查询数组中指定元素时，如果查询的元素在数组中不止出现一次，那么 array_search()函数将返回第一个匹配的键名。

如果想要返回所有匹配的键名，则可以使用 array_keys()函数，并设置可选参数 search_value。

array_keys()函数返回数组中所有的键名。语法如下：

```
array array_keys ( array input [, mixed search_value [, bool strict]] )
```

array_keys()返回 input 数组中的数字或者字符串的键名。

如果指定可选参数 search_value，则只返回该值的键名；否则 input 数组中的所有键名都会被返回。

专家点评

自 PHP 5 起，可以用 strict 参数来进行全等比较（===）。

问题 39　为什么$array[str]错了？

问题阐述

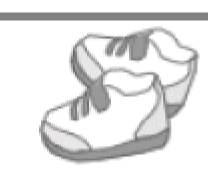
如果使用标识符[]定义数组时，那么应在用字符串表示的数组索引上加引号。例如，用$array['str']而不是$array[str]。

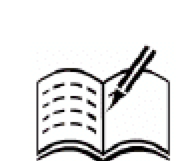
但是为什么$array[str]错了呢？例如下面的一段代码：

```
<?php
$array[str] = 'PHP 编程宝典';                    //定义数组元素
echo $array[str];                               //输出元素值
?>
```

专家解答

这样编写的代码是错的，但可以正常运行。那么为什么错了呢？原因是此代码中有一个未定义的常量（str）而不是字符串（'str'，注意引号）。如果 PHP 在以后定义此常量，而不幸的是你的代码中有同样的名字。它能够运行，是因为 PHP 自动将裸字符串（没有引号的字符串且不对应于任何已知符号）转换成一个其值为该裸字符串的正常字符串。例如，如果没有常量定义为 str，则 PHP 将把它替代为'str'并使用。





专家点评

这并不意味着总需要为键名加上引号。如果键名是常量或变量，则不需要加引号，否则会使 PHP 不能解析它们。

问题 40　如何设置 PHP 的错误级别？

问题阐述

如何配置 PHP 的错误级别？

专家解答

PHP 错误级别的设置有两种方法：

一种是直接操作 php.ini 配置文件，定位到“error_reporting =”配置选项，直接设置它的级别。

另一种是在程序中通过 error_reporting()函数设置错误级别。error_reporting()函数的语法如下：

```
int error_reporting ( [int level] )
```

参数 level 指定报错的级别。如果参数 level 未指定，则当前报错级别将被返回。表 2.2 给出的是参数 level 的可选值。

表 2.2　参数 level 的可选值

值	常　　量	说　　明
1	E_ERROR	报告致命性的运行错误
2	E_WARNING	报告运行时警告（非致命性错误）
4	E_PARSE	报告编译时解析错误
8	E_NOTICE	报告运行时提醒（此类错误经常是由于代码中的bug引起的，也可能是有意的行为造成的，如未初始化变量）
16	E_CORE_ERROR	PHP启动时初始化过程中的致命错误
32	E_CORE_WARNING	PHP启动时初始化过程中的警告（非致命性错误）
64	E_COMPILE_ERROR	编译时致命性错误
128	E_COMPILE_WARNING	编译时警告（非致命性错误）
256	E_USER_ERROR	用户自定义的错误信息
512	E_USER_WARNING	用户自定义的警告信息
1024	E_USER_NOTICE	用户自定义的提醒信息
2048	E_STRICT	编码标准化警告，允许PHP建议如何修改代码以确保最佳的互操作性和向前兼容性，即可移植性
4096	E_RECOVERABLE_ERROR	与E_ERROR类似，报告所有致命性的运行错误
8191	E_ALL	报告所有的错误和警告（不包括E_STRICT）

例如，下面的代码关闭了用户自定义的错误和警告，执行了某些操作，然后恢复到原始的报错级别。代码如下：

```
<?php
//禁用错误报告
```



```
error_reporting(0);
//报告运行时错误
error_reporting(E_ERROR | E_WARNING | E_PARSE);
//报告所有错误
error_reporting(E_ALL);
?>
```

Note

专家点评

任意数目的以上选项都可以用“或”来连接（用 OR 或 |），这样可以报告所有需要的各级别错误。

问题 41　PHP 有哪几种注释形式?

问题阐述

程序中的注释要用简单、明了、精确的语言表达出程序的处理说明。避免每行程序都使用注释，可以在一段程序的前面加一段注释，具有明确的处理逻辑，那么，PHP 中都有哪几种注释形式呢？

专家解答

1. 文件注释

每个包含 PHP 代码的文件必须至少在文件顶部包含这些 phpDocumentor 标签：

```
/**
 * 文件的简短描述
 *
 * 文件的详细描述（如果有的话）... ...
 *
 * LICENSE: 一些 license 信息
 *
 * @copyright   2005 Zend Technologies
 * @license     http://www.zend.com/license/3_0.txt     PHP License 3.0
 * @version     $Id:$
 * @link        http://dev.zend.com/package/PackageName
 * @since       File available since Release 1.2.0
*/
```

2. 类注释

每个类必须至少包含这些 phpDocumentor 标签：

```
/**
 * 类的简述
 *
 * 类的详细描述 （如果有的话）... ...
 *
 * @copyright   2005 Zend Technologies
 * @license     http://www.zend.com/license/3_0.txt     PHP License 3.0
 * @version     Release: @package_version@
```



```
* @link          http://dev.zend.com/package/PackageName
* @since         Class available since Release 1.2.0
* @deprecated Class deprecated in Release 2.0.0
*/
```



3. 函数注释

每个函数，包括对象方法，都必须有最少包含下列内容的文档块（docblock）：

函数的描述

所有参数

所有可能的返回值

因为访问级已经通过"public"、"private"或"protected"声明，不需要使用"@access"。

如果函数/方法抛出一个异常，则使用@throws：

```
@throws exceptionclass [description]
```

4. 块注释

块注释通常用于提供对文件、方法、数据结构和算法的描述，它被置于每个文件的开始处以及每个方法之前。另外，块注释也可以被用于其他地方，如方法内部。在功能和方法内部的块注释应该和它们所描述的代码具有一样的缩进格式。

块注释之首应该有一个空行，用于把块注释和代码分割开来，例如：

```
/*
* 这里是块注释
*/
```

5. 单行注释

短注释可以显示在一行内，并与其后的代码具有相同的缩进层级。如果一个注释不能在一行内写完，就应该采用块注释。单行注释之前应该有一个空行。单行注释的例子：

```
if (condition) {
/* 以下代码运行的条件 */
...
}
```

6. 尾端注释

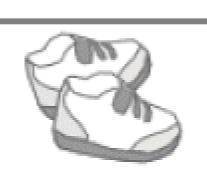
极短的注释可以与它们所要描述的代码位于同一行，但是应该有足够的空白来分开代码和注释。若有多个短注释出现于大段代码中，它们应具有相同的缩进。尾端注释的例子：

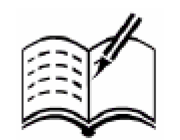
```
if ($a == 2) {
return TRUE; /* 对单一条件的说明 */
} Else {
return isPrime($a); /* 其余的条件 */
}
```

7. 行末注释

注释界定符“//”可以注释掉整行或一行中的一部分。例如：

```
if ($Euspice > 1) {
    ...     //注释用法
}
```





Note

专家点评

注释虽然必不可少，但也不应过多，不要被动地为写注释而写注释。

问题42　如何使用empty()函数判断文本框是否为空？

问题阐述

在网页程序的很多数据提交页面中，都不允许文本框内容为空，例如，在用户登录页面中，用户名和密码不允许为空等。除了在客户端通过JavaScript脚本控制文本框不为空之外，是否可以在服务器端进行判断呢？

专家解答

1. 解决思路

在服务器端判断表单提交的值是否为空，可以用empty()函数来完成。empty()函数检查变量是否为空，如果变量是空或零，则返回true。当变量为：""、0、"0"、null、false、array()、var $var以及没有任何属性的对象时都将被认为是空的。empty()函数的语法如下：

```
bool empty ( mixed var)
```

参数var为输入的变量。

2. 应用示例

应用empty()函数验证会员登录页面中的用户名和密码是否为空，如图2.1所示。当“用户名”和“密码”文本框为空时，单击“登录”按钮，对提交的表单信息进行验证，运行结果如图2.2所示。



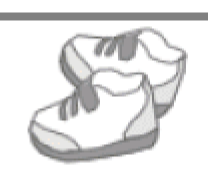
图2.1　用户登录页面



图2.2　文本框为空时显示的信息

本示例的关键代码如下：

```
<?php
$user = $_POST['user'];
$password = $_POST['password'];
if(empty($user) || empty($password)){
    echo "用户名或密码不能为空 <br /> <a href='login.html'>返回</a>";
}else{
    echo "输入的用户名为：$user 密码为：$password";
}
?>
```



问题 43 如何使用 strlen()函数计算文本长度？

问题阐述

在对超长文本进行显示时，由于页面空间有限，所以通常对其进行分页显示，在分页前，首先需要计算文本的长度，然后再对文本进行分页处理。

专家解答

1. 解决思路

获取指定字符串的长度使用 strlen()函数，其语法格式如下：

```
int strlen(string str)
```

参数 str 指定统计的字符串。

2. 应用示例

实现超长文本的分页显示，其中通过 strlen()函数获取文本的长度，同时还应用了文件系统函数库中的 fopen()函数、fgets()函数和字符串处理函数库中的 substr()函数，以及数学函数库中的 ceil()函数。其运行结果如图 2.3 所示。

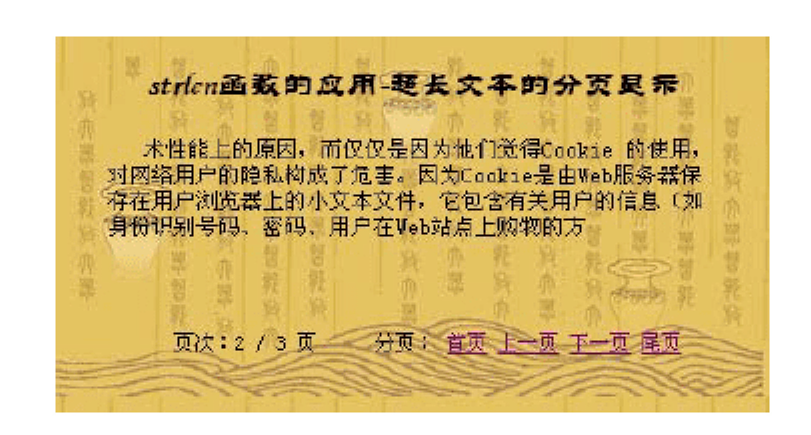


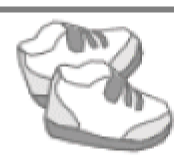
图 2.3 应用 strlen()函数实现超长文本的分页显示

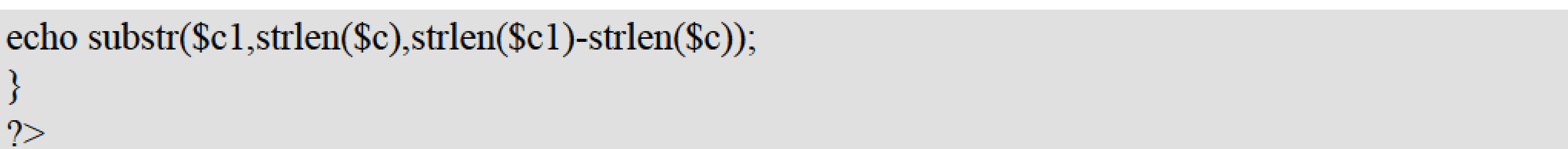
关键代码如下：

```
<?php
function msubstr($str,$start,$len){
$strlen=$start+$len;
    for($i=0;$i<$strlen;$i++) {
        if(ord(substr($str,$i,1))>0xa0) {
            $tmpstr.=substr($str,$i,2);
            $i++;
         }else{
    $tmpstr.=substr($str,$i,1); }
    }
    return $tmpstr;
}
?>
```

读取文本文件中的数据，应用 strlen()函数获取数据中字符串的长度进行输出，代码如下：

```
<?php
if($page){
$fp=fopen("file/file.txt","r");
$counter=fgets($fp);
    $length=strlen($counter);
    $page_count=ceil($length/200);
    $fp=fopen("file/file.txt","r");
    include("function.php");
    $c=msubstr($counter,0,($page-1)*200);
    $c1=msubstr($counter,0,$page*200);
```






```
echo substr($c1,strlen($c),strlen($c1)-strlen($c));
}
?>
```



Note

问题 44　如何使用 copy()函数对文件进行复制？

问题阐述

文件复制是将某一文件由一个路径复制到另一个路径下，通常我们会在 Windows 系统下完成这项操作。那么如何使用 copy()函数来实现这一功能呢？

专家解答

1. 解决思路

对文件进行复制操作可以使用 copy()函数，如果操作成功返回 true，否则返回 false。该函数的语法如下：

```
bool copy( string source, string dest)
```

参数 source 为必要参数，指定进行复制操作的文件目录及文件名称；参数 dest 也为必要参数，指定复制后文件的存储目录及名称。

2. 应用示例

应用 copy()函数实现将用户所选择的文件复制到其他位置。其关键代码如下：

```
<?php
 if($_POST["submit"]!=""){
     $filename=$_POST["filename"];
     $objfilename=$_POST["objfilename"];
     if(@copy($filename,$objfilename)){              //对文件进行复制操作
         echo "<script>alert('文件复制成功!');</script>";
     }else{
         echo "<script>alert('文件复制失败!');</script>";
     }
 }
?>
```

问题 45　如何在浏览器中直接输出图像？（fgetc()函数）

问题阐述

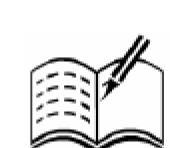
当选择一个图片文件后，如何在网页中直接查看图像呢？

专家解答

1. 解决思路

在浏览器中直接输出图像可以使用 fgetc()函数。该函数主要用于获取文件指针处的字符并返回，如果当前指针在文件尾则返回 false。语法如下：






```
string fgetc(resource handle)
```

参数 handle 指某文件的标识指针。

2. 应用示例

应用 fgetc()函数可以将图像直接在浏览器中进行显示。运行本示例，如图 2.4 所示，首先选择要显示的图像，然后单击“确定”按钮，即可在浏览器中显示出该图像。实例运行结果如图 2.5 所示。



图 2.4　应用 fgetc()函数输出图像



图 2.5　图像输出结果

关键代码如下：

```
<?php
    $filename=$_POST["filename"];
    $extendname=strstr($filename,".");
    if(!($extendname==".gif"||$extendname==".jpg"||$extendname==".bmp"||$extendname==".png")){
     echo "<script>alert('不支持这种格式的图像!');history.back();</script>";
     exit;          //判断是否为常规图像格式
    }
    $fp=fopen($filename,"r+");
     while (false !== ($char = fgetc($fp))){
        echo "$char";
    }
?>
```

问题 46　如何关闭危险函数？

问题阐述

如果打开了安全模式，那么函数禁止是可以不需要的，但是为了安全还是将其考虑进去。例如，不希望执行包括 system()等能够执行命令的 php 函数，或者能够查看 php 信息的 phpinfo()等函数。那么该如何禁止它们呢？

专家解答

```
disable_functions = system,passthru,exec,shell_exec,popen,phpinfo
```

如果要禁止任何文件和目录的操作，需要关闭很多文件操作：

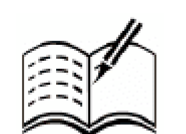
```
disable_functions = chdir,chroot,dir,getcwd,opendir,readdir,scandir,fopen,unlink,delete,copy,mkdir,rmdir, rename,file,file_get_contents,fputs,fwrite,chgrp,chmod,chown
```

以上只是列举了部分文件处理函数，也可以把上面执行的命令函数和该函数结合，就能够



抵制大部分的 phpshell 了。

如果将上述这两部分综合到一起，可以很好地控制用户的权限，保证 Web 项目的安全。



问题 47　如何将数组中的元素合成字符串？

问题阐述

如何将数组中的元素合成字符串？

专家解答

既然可以对字符串进行分割，返回数组，那么就一定可以对数组进行合成，返回一个字符串。这就是 implode()函数，将数组中的元素组合成一个新字符串。语法如下：

```
string implode(string glue, array pieces)
```

参数 glue 是字符串类型，指定分隔符；参数 pieces 是数组类型，指定要被合并的数组。

例如，应用 implode()函数将数组中的内容以*为分隔符进行连接，从而组合成一个新的字符串，代码如下：

```
<?php
$str="PHP 编程宝典*NET 编程宝典*ASP 编程宝典*JSP 编程宝典";    //定义字符串常量
$str_arr=explode("*",$str);                                 //应用标识*分割字符串
$array=implode("*",$str_arr);                               //将数组合成字符串
echo $array;                                                //输出字符串
?>
```

运行结果为：PHP 编程宝典*NET 编程宝典*ASP 编程宝典*JSP 编程宝典

问题 48　如何解决使用 empty 检查函数返回结果时报错？

问题阐述

在 PHP 开发时，当使用 empty 检查一个函数返回的结果时会报错：Fatal error: Can't use function return value in write context。例如：

```
<?php
echo empty(strlen('be-evil.org'));
?>
```

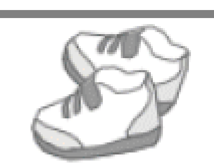
专家解答

查看 PHP 手册，在 empty 函数描述的地方有以下文字：

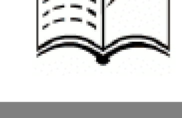
> Note: empty() only checks variables as anything else will result in a parse error. In other words, the following will not work: empty(trim($name)).

empty() 只检测变量，检测任何非变量的东西都将导致解析错误!

因此，我们不能用 empty()函数来直接检测函数返回的值，上面例子的解决方案如下：



```
<?php
$length = strlen('be-evil.org');                    //将返回值定义到变量中
echo empty($length);                                //检测变量
?>
```



问题 49 如何使用 mb_substr 函数截取中文字符串?

问题阐述

通过 mb_substr 函数可以对中文字符串进行截取，并且不会出现乱码的问题。但是应用该函数还有一些前期的工作要做，那么具体都有哪些工作呢？

专家解答

（1）首先，确保 Windows/system32 下有 php_mbstring.dll 文件，如果没有，可以从 php 安装目录 extensions 中复制一个到 Windows/system32 中。

（2）在 Windows 目录下找到 php.ini 文件，在该文件中搜索 mbstring.dll，找到下面的语句：

```
;extension=php_mbstring.dll
```

把前面的;号去掉，这样 mb_substr 函数就可以生效了。

mb_substr()函数实现对指定的字符串进行截取操作，其语法如下：

```
string mb_substr ( string str, int start [, int length [, string encoding]] )
```

参数说明：

❶ str：指定被截取的字符串。

❷ start：指定截取的开始位置。

❸ length：指定截取的长度。

❹ encoding：指定被截取字符串的编码格式。

例如，应用 mb_substr()函数对指定的字符串进行截取。

```
<?php
$str = '这样一来我的字符串就不会有乱码^_^';
echo mb_substr($str,'utf-8');
?>
```

运行结果为：这样一来我的字符串就不会有乱码^_^

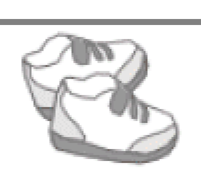
问题 50 如何解决 PHP 无法执行 system()函数的问题?

问题阐述

PHP 无法执行 system()函数是什么原因呢？

专家解答

PHP 无法执行 system()函数，可能有以下几种情况：





（1）Warning: system() has been disabled for security reasons

该错误是由于服务器安全配置将 system 函数关闭了，打开 php.ini 文件，将 disable_functions 等号后面的 system 去掉即可。

（2）Warning: system() [function.system]: Unable to fork

该错误是由于当前站点的运行账号无法访问 cmd.exe，找到 Windows\system32\cmd.exe 属性，添加当前站点的运行账号权限或 users 组权限即可。

（3）在 Web 访问下执行没有任何响应也不报错，使用 php cli 命令行执行却可以正常返回结果，该错误一般是由 Windows 系统路径引起的，程序中可能是使用了绝对路径去访问程序，例如：

```
system('D:\server\DLL\IECapt.exe --url=http://tech.cncms.com --out="D:\web\tt.fei.cn\htdocs\ttt\tech.png"');
```

此时用命令行执行结果正确，但用 Web 访问就会返回空白页，没有错误。

修改方法：将文件的访问路径改为相对路径，并将 IECapt.exe 复制到站点程序当前目录下，并将代码改为：

```
system('IECapt.exe --url=http://www.baidu.com --out=t/tech.png');
```

再次访问页面，就可以正常输出了。

问题 51　如何对查询关键字进行描红？

问题阐述

很多网站都具有查询功能，使用该功能可以让用户很快地找到自己所需要的内容，但传统的搜索只是将符合条件的记录显示出来，这样使用户很难在记录中找到自己需要的关键技术，所以，为了满足用户的需求，就产生了关键字描红技术。那么该技术是如何实现的呢？

专家解答

1. 解决思路

对查询关键字进行描述可以使用 str_replace()函数来替换指定字符串，从而实现关键字的描红技术。str_replace()函数执行字符串替换操作。其语法如下：

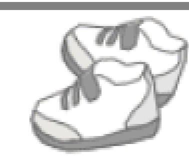
```
mixed str_replace ( mixed search, mixed replace, mixed subject [, int &count] )
```

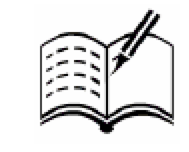
参数说明：

❶ search：必要参数，指定需要查找的字符串。
❷ replace：必要参数，指定替换的值。
❸ subject：必要参数，指定的查找范围。
❹ count：可选参数，获取执行替换的数量。

2. 应用示例

本例实现了一个简单的搜索功能，输入要搜索的关键字，单击“搜索”按钮，将通过 POST 方法接收到数据。首先进行描红处理，其次应用 str_replace()函数对指定字符串信息的指定内容进行替换。运行结果如图 2.6 所示。





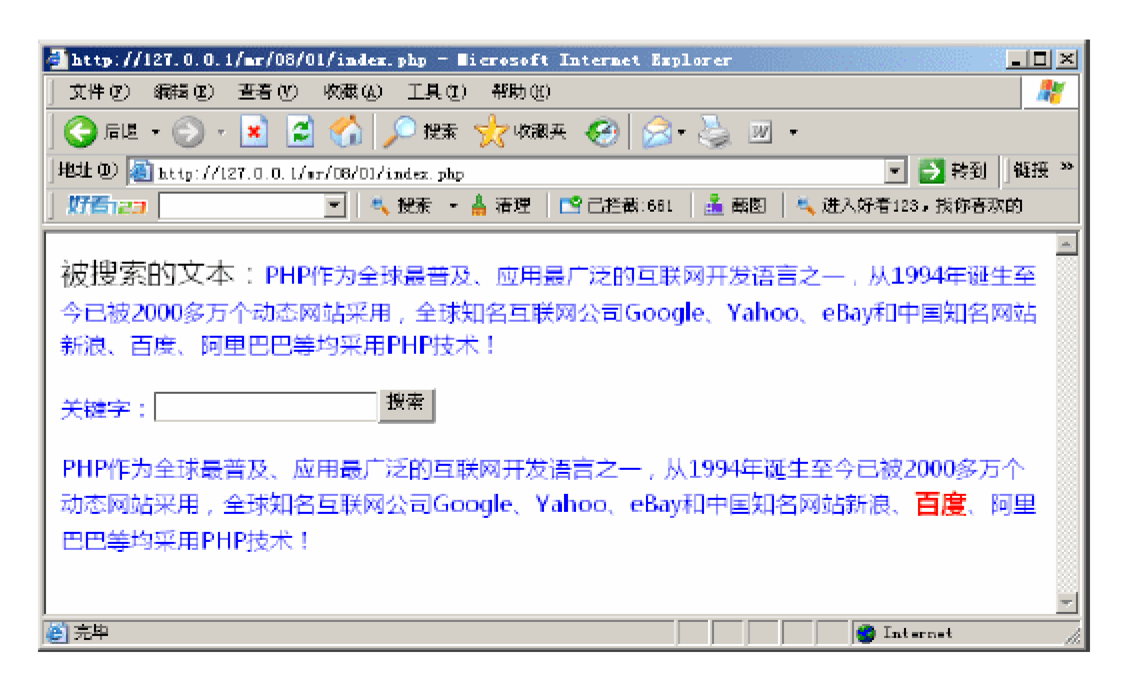


图 2.6　对查询关键字进行描述

关键代码如下：

```
<?php
    if($_POST[sub]){
        $a = "<b style='color:red;font-size:18px;'>".$_POST[text]."</b>";
        echo str_replace("$_POST[text]",$a,$str);
    }
?>
```

专家点评

对查询关键字进行描红在网站搜索中经常用到，实现这一功能并不复杂，希望读者可以熟练掌握。

问题 52　如何对字符串与 HTML 进行转换？

问题阐述

字符串与 HTML 之间的相互转换是非常简单的，通过转换可以极大地方便 Web 浏览器的浏览和字符串的输出。

专家解答

1. 解决思路

该技术的实现主要应用 htmlentities()和 htmlspecialchars()函数。htmlentities()函数将所有的字符串都转换成 HTML 字符串，而 htmlspecialchars()函数只是将字符串中的某些特殊字符转换成 HTML 格式。

htmlentities()函数的语法如下：

```
string htmlentities(string string,[int quote_style],[string charset])
```

参数说明如表 2.3 所示。

说明：

在应用 htmlentities()函数对字符串到 HTML 的转换过程中，设置参数 charset 的字符集为“GB2312”，可以避免在输出中文字符时出现乱码的问题。其前提是页面是 gb2312 编码，如果页面是 utf-8 编码格式，那么这个值就要设置为 utf-8。

表 2.3　htmlentities()函数的参数说明

参　　数	说　　明
string	必要参数，指定要转换的字符串
quote_style	可选参数，选择如何处理字符串中的引号，有3个可选值，（1）NET_COMPAT，转换双引号，忽略单引号，它是默认值；（2）ENT_NOQUOTES，忽略双引号和单引号；（3）ENT_QUOTES，转换双引号和单引号
charset	可选参数，确定转换所使用的字符集，默认字符集是“ISO-8859-1”，指定字符集后就能够避免转换中文字符出现乱码的问题


htmlspecialchars()函数的语法如下：

```
string htmlspecialchars(string string)
```

该函数将一些特殊字符转换成 HTML 格式，而不会将所有字符都转换成 HTML 格式。该函数转换的特殊字符如表 2.4 所示。

表 2.4　htmlspecialchars()函数转换的特殊字符

参　　数	说　　明
&（和）	转成 &
" "（双引号）	转成 "
>（大于号）	转成 >
<（小于号）	转成 <

2. 应用示例

下面应用 htmlentities()函数将字符串转换成 HTML 字符串，并且设置转换的字符集为“GB2312”，避免中文输出乱码的问题。运行结果如图 2.7 所示。

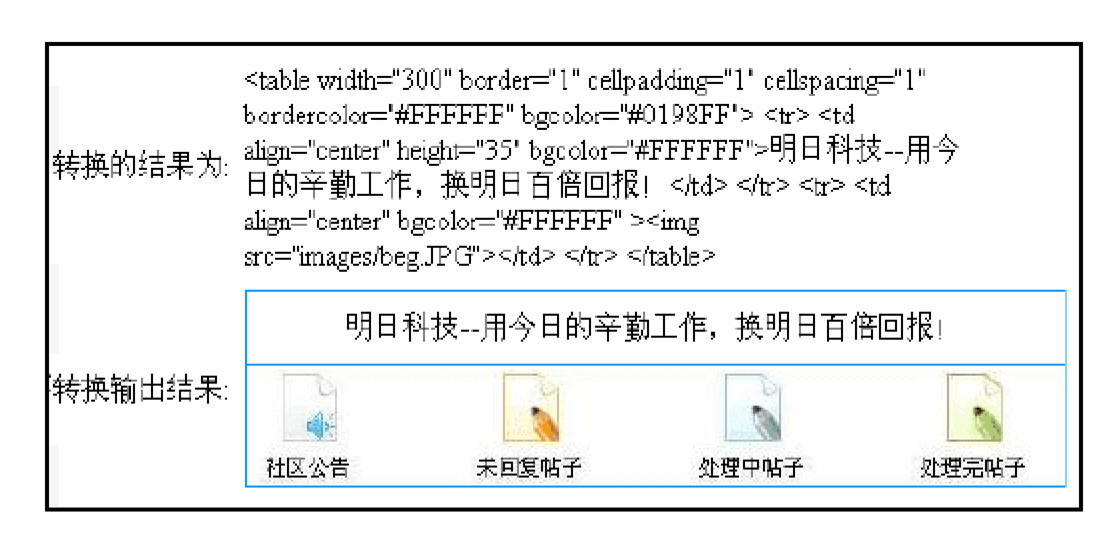
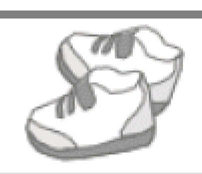


图 2.7　将字符转换为 HTML

关键代码如下：

```
<?php
    $str='<table width="300" border="1" cellpadding="1" cellspacing="1" bordercolor="#FFFFFF" bgcolor="#0198FF">
        <tr>
          <td align="center" height="35"  bgcolor="#FFFFFF">明日科技--用今日的辛勤工作，换明日百倍回报！ </td>
        </tr>
        <tr>
          <td align="center" bgcolor="#FFFFFF" ><img src="images/beg.JPG"></td>
        </tr>
```



```
        </table>';
        echo htmlentities($str,ENT_QUOTES,"utf-8")."<br>";    //设置转换的字符集为"GBK"
        ?>
    <?php echo $str; ?>
```



专家点评

字符串与 HTML 相互转换的技术应用最广泛的地方就是输出数据库中的数据，例如，当数据库中存储的是一个可以运行程序的代码时，如果没有对该数据进行转换而直接输出，那么输出的结果将是该程序的运行结果，而不是程序代码。目前在论坛、博客（帖子浏览），特别与编程有关的论坛或博客中应用最为广泛。

问题 53　如何屏蔽违禁词语？

问题阐述

很多论坛、评论、贴吧都有过滤敏感词的功能。所谓的“敏感词”是指网站管理者为了正常经营发展、减少不必要的纷争以及消除不良影响，而人为设定的一些较易引起人们歧义思维的词汇。屏蔽违禁词主要是为了使网站的内容健康而规范，这项功能在论坛发表中比较常用。如果不设置违禁词屏蔽，用户随意发表不健康的内容，会对网站造成很大的不良影响。那么屏蔽违禁词语的功能是如何实现的呢？

专家解答

1. 解决思路

实现违禁词屏蔽功能的方法有很多种，例如，可以应用 file()函数读取存储在文本文件中的敏感词汇，并将其存储在$file_word 数组中。然后，应用 for 循环语句自动读取数组元素（敏感词）。最后通过正则表达式 preg_match()函数检验用户提交的留言信息中是否含有敏感词。

2. 应用示例

在 IE 浏览器中输入地址，按 Enter 键，在用户提交留言信息后，系统将签写的留言信息与本模块设定的敏感词进行一一对比，如果在用户签写的留言信息中存在敏感词，如“喜欢”，就会弹出留言中含有敏感词的警告信息，结果如图 2.8 所示。

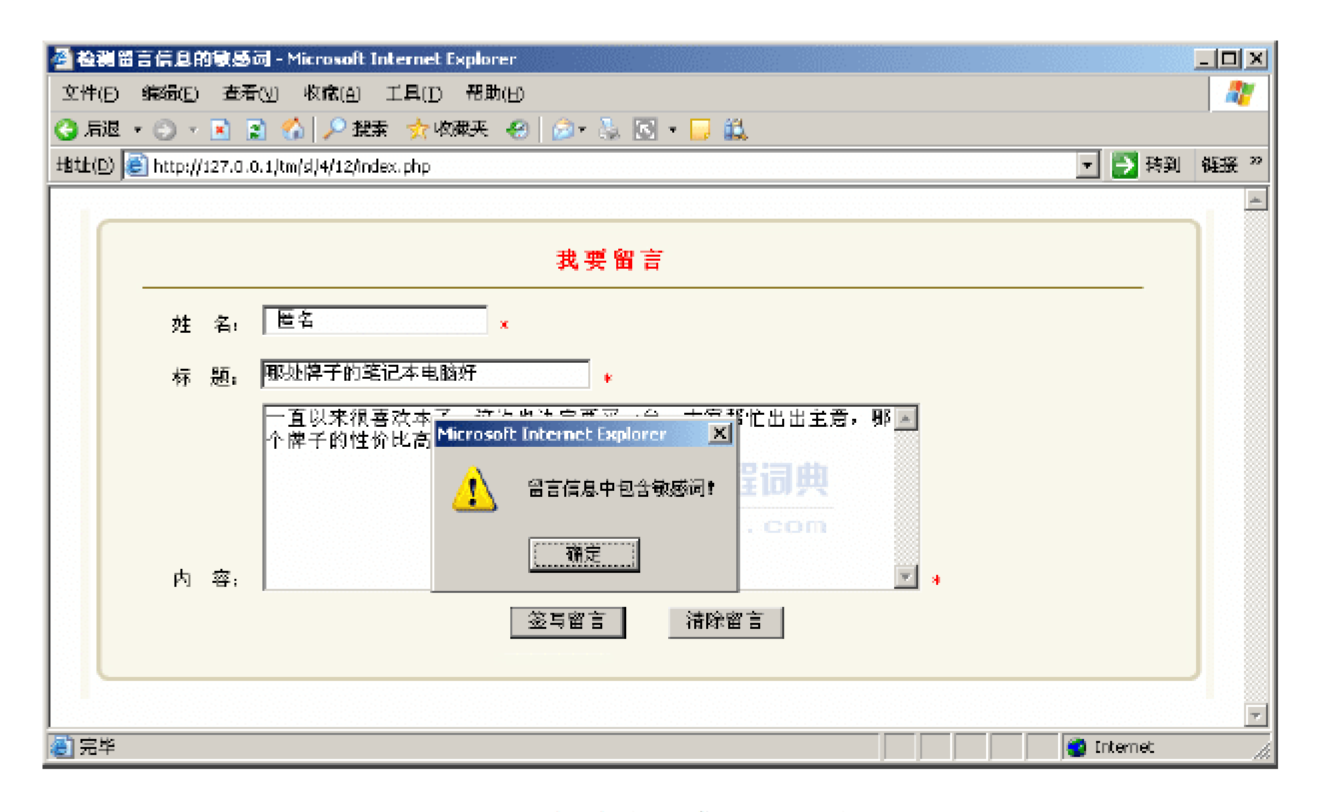
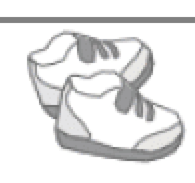


图 2.8　过滤敏感词“喜欢”



示例代码参考如下：

```
<?php
/*
    1．Language()函数，只接受文本框和文本域的两组信息。
    2．本函数需调用自定义函数 con()连接数据库。
    3．将违禁词语保存在数组中。
    4．通过正则表达式函数进行筛选。
*/
function language($strr,$strr1){                    //屏蔽标题信息
    con("localhost","root","111","db_database","utf-8");
        $array=array("喜欢","喜爱","笔记本电脑");
        for($a=0;$a<count($array);$a++){
            if(preg_match("/$array[$a]/",$strr)){
                echo "<script>alert('标题内容包含违禁词语或为空，请更改');</script>";
                break;
            }else{
                if($strr1!=""){
                    mysql_query("insert into tb_demo02(id,title,content)values('','$_POST[text]',
'$_POST[tear]')");
                    echo "<script>alert('发布成功');location.href='index.php?id=2'</script>";
                    break;
            }else{
                echo "<script>alert('发布内容为空');</script>";
                break;
            }
        }
    }
}
/*
自定义函数 conn()预期接收 5 个参数，其中包括数据源、数据库用户名和密码、连接的数据库
名称和页面的编码格式。
*/
    function con($local,$root,$pwd,$db,$code){
        $conn=mysql_connect($local,$root,$pwd) or die ("Connect MySQL False");
        mysql_select_db($db,$conn) or die ("Connect DB False");
        mysql_query($code);
    }
?>
```

专家点评

用户在论坛系统中发表主题时，如果使用了不恰当的词语，系统往往会提示用户用了违禁的词语，因此发表没有成功，请更改等；或者还有些用户尽管系统提示发表成功，但是在浏览信息时发现发表的信息内容被系统自动屏蔽为不可见的模式了。

问题 54　如何对字符串进行分割及连接？

问题阐述

字符串的连接与分割是非常重要的两个内容，通过其可以将数组按照指定的规则转换成字





符串，也可以将字符串按照指定的规则进行分割，返回一个数组。其应用范围非常广，例如网站的购物车或在线投票系统等。那么这两个功能具体是如何实现的呢？

专家解答

1. 解决思路

对字符串进行分割或连接应用的是 implode()和 explode()函数。

（1）explode()函数，按照指定的规则对一个字符串进行分割，返回值为数组。语法如下：

```
array explode ( string separator, string string [, int limit]);
```

参数 string string 表示分割条件；参数 separator 表示指定的分隔符；参数 int limit 表示指定生成数组的最大值。

（2）implode()函数，与 explode 函数配合使用，将数组的内容组合成一个字符串。其语法如下：

```
string implode ( string glue, array pieces);
```

参数 glue 是字符串类型，指定分隔符；参数 pieces 是数组类型，指定要被合并的数组。

2. 应用示例

（1）字符串的分割

本示例开发一个在线投票管理系统，通过向文本框中输入要进行投票的内容，并且以“*”作为每项内容之间的分割符。单击“提交”按钮，将在投票选项中的内容进行输出，运行结果如图 2.9 所示。

关键代码如下：

```
<?php
$str1=$_POST[str1];
$str2=$_POST[str2];
$array=explode($str2,$str1);
echo "该字符串分割结果为：".""<br>";
while(list($name,$value)=each($array)){
   echo $value."<br>";
 }
?>
```

（2）字符串的连接

implode()函数将数组按照指定的规则转换成一个字符串，实现根据数据库中的数据表动态地生成数据录入的页面，其中主要应用 implode()函数将表单中提交的数据存储到数据库中。运行结果如图 2.10 所示。

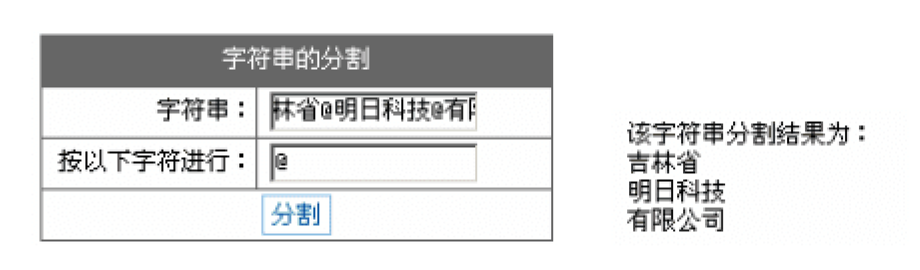
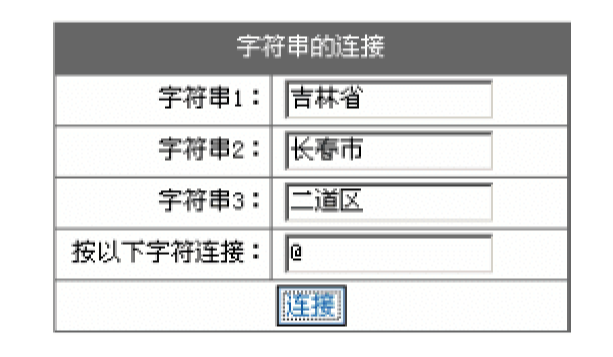


图 2.9　字符串的分割

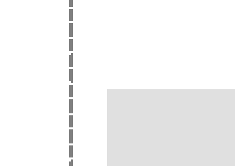


该字符串合并结果为：
吉林省@长春市@二道区

图 2.10　字符串的连接

关键代码如下：





Note

```
<?php
$str1[0]=$_POST[str11];
$str1[1]=$_POST[str12];
$str1[2]=$_POST[str13];
$str2=$_POST[str2];
$str=implode($str2,$str1);
echo "该字符串合并结果为：".""<br>";
echo $str;
?>
```

问题 55　如何将 IP 地址转换成对应的数字？

问题阐述

IP 地址是互联网身份的象征，通过它可以得到用户的相关信息。很多网站都会记录访问者的 IP 地址，通过它可以判断用户的所在地，这样就可以了解每个地区用户的访问量了。

专家解答

1. 解决思路

通过 PHP 中提供的$REMOTE_ADDR 变量可以获取用户的 IP 地址，但并不能判断 IP 的所在地，为了解决上述问题可以先将该 IP 转换成其对应的数字，之后根据该数值查找 IP 所在地。

首先需要验证输入的 IP 地址是否合法，如果合法，再将其转换为对应的数值。验证 IP 地址是否合法可以通过正则表达式实现。

IP 地址使用 4 位地址定位设备，它采用点分十进制数的格式“xxx.xxx.xxx.xxx”，如 192.168.1.66。因此，验证 IP 地址是否合法的正则表达式如下：

```
/(\d+).(\d+).(\d+).(\d+)/
```

将 IP 地址转换为对应的数值可以应用以下公式实现：

```
P1*255*255*255+P2*255*255+P3*255+P4*1
```

在上面的公式中，P1 代表 IP 地址第 1 段的数值；P2 代表 IP 地址第 2 段的数值；P3 代表 IP 地址第 3 段的数值；P4 代表 IP 地址第 4 段的数值。

2. 应用示例

下面通过正则表达式验证 IP 地址是否合法，然后将 IP 地址转换为对应的数字，进而获取 IP 地址所在地。运行结果如图 2.11 所示。

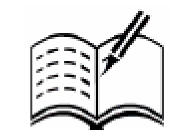


图 2.11　将 IP 地址转换为对应的数字

关键代码如下：

```
//验证 IP 地址
<script language="javascript">
function deal(myform){
      var ip=form1.ip.value;
      objExp=/(\d+).(\d+).(\d+).(\d+)/;
      if(ip==""){
            alert("请输入 IP 地址!");form1.ip.focus();return;
```





```
        }
        if(objExp.test(ip)){
            myform.ipNum.value=convert(ip);
        }else{
            alert("您输入的 IP 地址不合法!");form1.ip.focus();return;
        }
    }
    </script>
    //将 IP 地址转换为对应的数字
    <script language="javascript">
    function convert(ip){
        ip=ip.split(".");
        vip=ip[0]*255*255*255+ip[1]*255*255+ip[2]*255+ip[3]*1;
        return vip;
    }
    </script>
```

专家点评

将 IP 地址转换为对应的数字后，如果要查看对应的地区，可以到网上查找。

问题 56　如何创建新型字符串?

问题阐述

以往字符串的定义方式都是将字符串用单引号或双引号括起来进行创建的，在 PHP 5.0 以后的版本中出现了一种新型字符串的定义方式，下面进行介绍。

专家解答

1. 解决思路

创建新型字符串以“<<<”开始，后面紧跟着字符串开始标记。新型字符串格式如下：

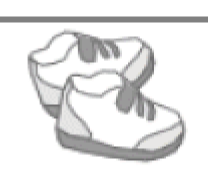
```
<<<标记
    字符串内容        //相当于"字符串内容"
  标记;
```

2. 应用示例

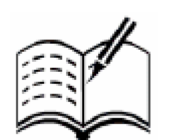
创建 PHP 5 新型字符串，本例以 strmark 作为标记为例，之后为字符串的内容，最后以标记加分号“strmark;”结束。代码如下：

```
<?php
$str=<<<strmark
吉林省明日科技
strmark;
echo $str;
?>
```

运行本示例，如果在浏览器中显示“吉林省明日科技”，则说明程序创建成功。



问题 57　如何对字符串进行加密解密？



Note

问题阐述

对于电子商务网站的用户来说，最重要的就是账号和密码的安全。如果账号和密码未经过加密处理，则大大增加了安全隐患，给非法用户以可乘之机。为了确保用户的利益及隐私不被泄露，多数网站开始对用户的账号和密码进行加密。

专家解答

1. 解决思路

字符串的 base64 编码与解码可以通过 PHP 的预定义函数 base64_encode()和 base64_decode()来实现。对字符的 base64 编码用函数 base64_encode()实现，语法如下：

```
string base64_encode(string str)
```

该函数实现对字符串 str 的 base64 编码。

对已经进行 base64 编码的字符串进行解码用函数 base64_decode()实现，语法如下：

```
string base64_decode(string str)
```

该函数实现对已经进行 base64 编码的字符串进行解密。

2. 应用示例

通过字符串函数 base64_encode()、base64_decode()实现对密码的加密和解密，运行结果如图 2.12 所示。



图 2.12　用户注册的密码进行加密和解密

创建 PHP 脚本，当单击“提交”按钮时，通过字符串函数 base64_encode()、base64_decode()实现对字符串的加密和解密，代码如下：

```
<?php
        if($_POST[sub]){                                        //通过 POST 方法接收参数
            echo "经加密<hr>";
            echo "用户名：".$_POST[text];
            $str = base64_encode($_POST[pwd]);                  //对密码进行加密
            echo "密码：". $str;
            echo "经解密";
            echo "<hr>";
            echo "用户名：".$_POST[text];
            echo "密码：". base64_decode($str);                 //对密码进行解密
        }
?>
```

专家点评

PHP 中提供了很多加密和解密的函数，例如 base64_encode()或 md5()等，在实际使用时，用户可以挑选一个合适的函数使用。





问题 58　如何比较字符串的大小?

问题阐述

在 Web 项目开发过程中，经常需要确定两个或多个字符串之间的大小关系。

专家解答

1. 解决思路

PHP 中实现字符串的大小比较可以通过 strcasecmp()函数和 strcmp()函数来实现。语法如下：

```
int strcasecmp(string str1, string str2)
int strcmp(string str1, string str2)
```

这两个函数用于比较两个字符串的大小（基于 ASCII 码值），如果 str1>str2 则返回 1；如果 str1<str2 则返回-1；如果 str1=str2 则返回 0。二者的区别在于函数 strcasecmp()在比较的过程中不区分大小写，而函数 strcmp()在比较的过程中区分大小写。

2. 应用示例

下面使用 strcasecmp()函数和 strcmp()函数来实现字符串大小的比较，运行结果如图 2.13 所示。

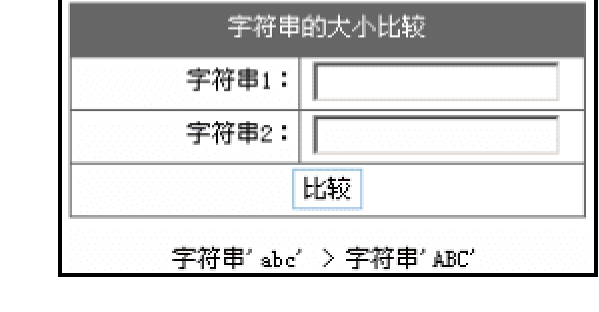


图 2.13　比较字符串的大小

关键代码如下：

```
<?php
if($_POST["submit"]!="")
 {
  $str1=$_POST[str1];
  $str2=$_POST[str2];
  $result=strcmp($str1,$str2);
  if($result==1)
    echo "<div align=center>字符串'".$str1."' > 字符串'".$str2."'</div>";
  if($result==0)
    echo "<div align=center>字符串'".$str1."' = 字符串'".$str2."'</div>";
  if($result==-1)
    echo "<div align=center>字符串'".$str1."' < 字符串'".$str2."'</div>";
  }
?>
```

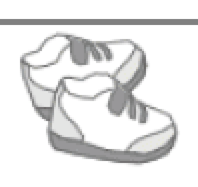
专家点评

比较字符串的大小主要集中在对类型及大小写的比较上，在实际应用中，当遇到同一个英文单词，如果其大小写不同，会使运算结果出错，就可以先对字符串进行比较，再根据情况做下一步的操作。

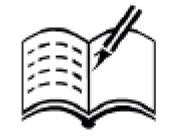
问题 59　如何获取上传文件的扩展名?

问题阐述

验证上传文件的格式类型是对上传文件的一种安全保障，大家不妨设想一下，对于实现任意



格式的文件上传，一方面会给非法用户以可乘之机，另一方面也给服务器带来很大的压力。可见对上传文件格式的规范是很有必要的，实现这一功能的前提需要先获取上传文件的扩展名。



专家解答

1. 解决思路

获取上传文件的扩展名可以应用 strstr()函数，从上传文件名称中的“.”开始截取，即可获取到上传文件的后缀名称，然后与程序中定义的可上传文件格式进行匹配，如果匹配成功，则表示该文件符合上传条件。

strstr()函数获取一个指定字符串在另一个字符串中首次出现的位置到后者末尾的子字符串。如果执行成功，则返回剩余字符串（存在相匹配的字符）；否则返回 false。语法如下：

```
string strstr ( string haystack, string needle)
```

参数 haystack 指定从哪个字符串中进行搜索；参数 needle 指定搜索的对象。如果该参数是一个数值，那么将搜索与这个数值的 ASCII 码相匹配的字符。

说明：

strstr()函数区分大小写，如果不需要对大小写加以区分，可以使用 stristr()函数。

2. 应用示例

本示例应用 strstr()函数获取上传图片的后缀，以此来判断上传图片的格式是否正确，如果正确则将图书上传到服务器根目录下的 upload 文件夹下，否则给出提示信息。运行结果如图 2.14 和图 2.15 所示。



图 2.14　应用 strstr()检索上传图片的后缀



图 2.15　应用 strstr()检索上传图片的后缀

其关键代码如下：

```
<form name="form" method="post" action="index.php" enctype="multipart/form-data">
    <input name="u_file" type="file" size="24"/>
    (<span class="STYLE1">*上传图片是.jpg 格式，大小不能超过 1.2MB</span>)
    <input type="image" name="imageField" src="imges/sc.bmp" onClick="form.submit();">
</form>
```



```
<?php
if($_FILES[u_file][name]==true){
        $file_path = "./upload\\";                    //定义图片在服务器中的存储位置
        $picture_name=$_FILES[u_file][name];          //获取上传图片的名称
        $picture_name=strstr($picture_name , ".");    //通过 strstr()函数截取上传图片的后缀
        if($picture_name!= ".jpg" && $picture_name!= ".JPG" ){    //根据后缀判断图片的格式
            echo  "<script>alert('上传图片格式不正确,请重新上传');  window.location.href=
'index.php';</script>";
        }else if($_FILES[u_file][tmp_name]){
            move_uploaded_file($_FILES[u_file][tmp_name],$file_path.$_FILES[u_file][name]);
                                                      //执行图片上传
            echo "<script>alert('图片上传成功!'); window.location.href='index.php';</script>";
        }else{
            echo "<script>alert('上传图片失败！ '); window.location.href='index.php';</script>";
        }
}
?>
```



问题 60　如何控制页面中输出字符串的长度?

问题阐述

在网页程序中，经常需要显示一些较长的信息，如公告信息、标题名称等。如果对这些信息进行全部输出，则页面中将会显示出长短不一的字符串信息，会使页面看起来比较杂乱。为了保持整个页面布局的合理、美观，经常需要对这些较长的字符串进行部分输出，以保证显示的字符串长度一致。对于超出长度部分，可以使用指定的字符进行替换。

专家解答

1. 解决思路

控制页面中输出字符串的长度可以使用 str_replace()函数和 substr()函数。str_replace()函数主要用于以指定的字符替换超出长度的部分；substr()函数主要实现截取字符的确定位置。

（1）str_replace()函数，实现字符串的替换。语法如下：

```
mixed str_replace ( mixed search, mixed replace, mixed subject , int &count )
```

str_replace()函数将所有在参数 subject 中出现的 search 以参数 replace 替换，参数&count 表示替换字符串执行的次数。

str_replace()函数的参数说明如表 2.5 所示。

表 2.5　str_replace()函数的参数说明

参　数	说　明
search	指定将要被替换的字符
replace	指定替换所使用的字符
subject	指定被操作的字符串
&count	替换字符串执行的次数


Note

（2）substr()函数，从指定的字符串中按照指定的位置截取一定长度的字符。语法如下：

```
string substr(string str,int start,int length)
```

substr()函数的参数说明如表 2.6 所示。

表 2.6　substr()函数的参数说明

参　数	说　明
str	指定字符串对象
start	指定开始截取字符串的位置。如果参数start为负数，则从字符串的末尾开始截取
length	可选参数，指定截取字符的个数，如果参数length为负数，则表示取到倒数第length个字符

说明：

本函数中参数 start 的指定位置是从 0 开始计算的，即字符串中的第一个字符表示为 0。

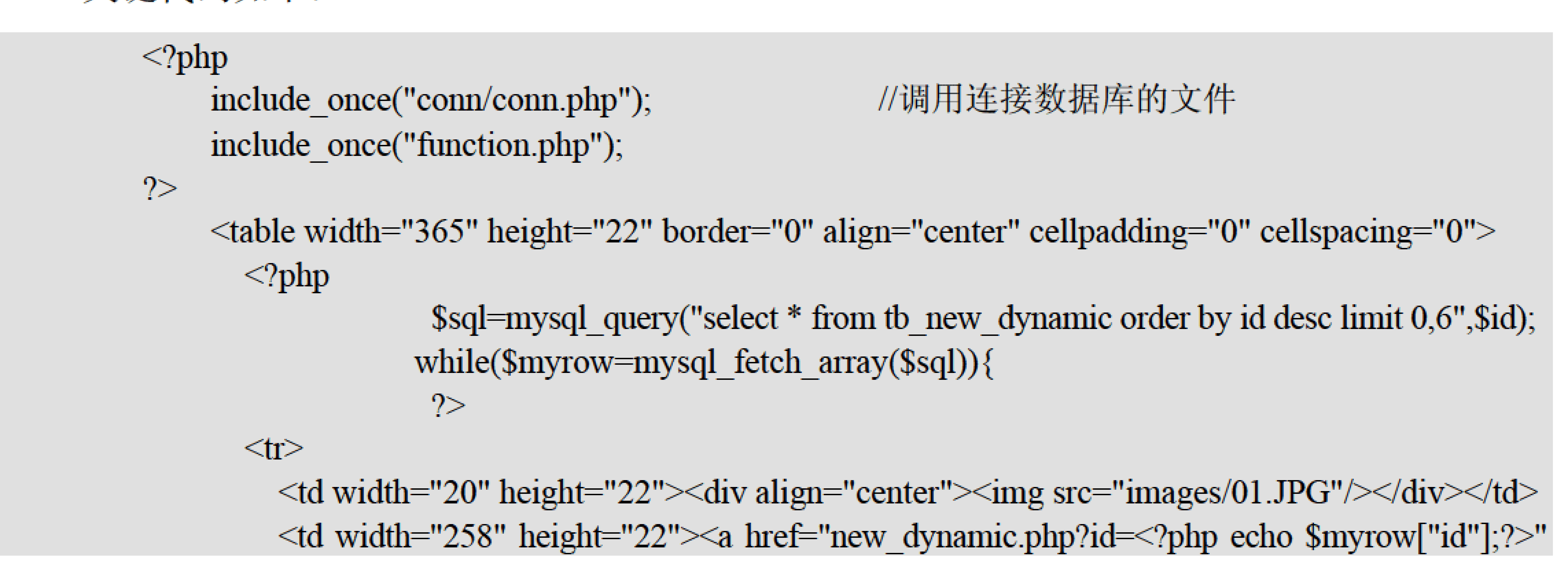
2. 应用示例

本例实现在输出最新动态标题时截取 24 个字节标题名称，超出长度的内容以省略号代替，运行结果如图 2.16 所示。



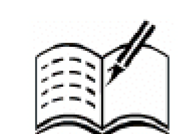
图 2.16　控制页面中输出字符串的长度

关键代码如下：



```
<?php
    include_once("conn/conn.php");                    //调用连接数据库的文件
    include_once("function.php");
?>
    <table width="365" height="22" border="0" align="center" cellpadding="0" cellspacing="0">
      <?php
             $sql=mysql_query("select * from tb_new_dynamic order by id desc limit 0,6",$id);
            while($myrow=mysql_fetch_array($sql)){
             ?>
      <tr>
        <td width="20" height="22"><div align="center"><img src="images/01.JPG"/></div></td>
        <td width="258" height="22"><a href="new_dynamic.php?id=<?php echo $myrow["id"];?>"
```





```
title="<?php echo $myrow["dynamic_title"];?>">
                    <?php
                                echo unhtml(msubstr($myrow["dynamic_title"],0,24));
                                 if(strlen($myrow["dynamic_title"])>24){
                                    echo " ... ";
                                 }
                      ?>
                  </a> </td>
                  <td width="87"><?php   echo "<font color=red>[".substr(str_replace("-","/",$myrow[createtime]),
0,10)."] </font>";?></td>
                </tr>
                <tr>
                  <td colspan="3"></td>
                </tr>
                <?php
                  }
                  ?>
              </table>
```

问题 61 如何定义上传文件的名称？

问题阐述

在上传文件时需要指定上传文件路径，但面对大量用户上传的不同文件，难免会有重名的可能，这样就会造成文件替换，从而导致文件丢失，所以对上传文件的名称进行重新定义是非常有必要的。

专家解答

1. 解决思路

通过 strstr()函数对上传文件名称进行截取，获取上传文件的后缀。

通过 time()函数获取系统当前时间戳，重新定义上传文件的名称。

应用 move_uploaded_file()函数实现文件上传。

在执行文件上传之前，为了防止潜在的攻击对原本不能通过脚本交互的文件进行非法管理，应用 is_uploaded_file()函数判断指定的文件是否是通过 HTTP POST 上传的，如果是则返回 True。

（1）is_uploaded_file()函数，判断指定的文件是否是通过 HTTP POST 上传的。其语法如下：

```
bool is_uploaded_file ( string filename )
```

参数 filename 必须指定类似于$_FILES['filename']['tmp_name']的变量，不可以使用从客户端上传的文件名$_FILES['filename']['name']。

通过 is_uploaded_file()函数对上传文件进行判断，可以确保恶意用户无法欺骗脚本去访问原本不能访问的文件，如/etc/passwd。

（2）move_uploaded_file()函数将文件上传到服务器中指定的位置。如果成功返回 True，否则返回 False。语法如下：

```
bool move_uploaded_file ( string filename, string destination )
```



参数 filename 指定上传文件的临时文件名，即$_FILES[tmp_name]；参数 destination 指文件上传后保存的新路径和名称。如果参数 filename 不是合法的上传文件，则不会出现任何操作，move_uploaded_file()将返回 False；如果参数 filename 是合法的上传文件，但出于某些原因无法移动，则也不会出现任何操作，move_uploaded_file()将返回 False，此外还会发出一条警告信息。

（3）rand()函数，使用其生成的随机整数作为上传文件的新名称。

Note

2. 应用示例

本示例重新定义上传文件的名称，选择要上传的文件，单击“提交”按钮，文件将上传到程序中指定的本地路径下的 upfile 文件夹中，并重新定义文件名称。运行结果如图 2.17 所示。



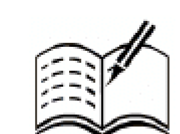
图 2.17　选择上传文件

其关键代码如下：

```
<?php
if(!empty($_FILES[up_picture][name])){          //判断上传内容是否为空
    if($_FILES['up_picture']['error']>0){       //判断文件是否可以上传到服务器
        echo "上传错误:";
        switch($_FILES['up_picture']['error']){
            case 1:
                echo "上传文件大小超出配置文件规定值";
            break;
            case 2:
                echo "上传文件大小超出表单中约定值";
            break;
            case 3:
                echo "上传文件不全";
            break;
            case 4:
                echo "没有上传文件";
            break;
        }
    }else{
        if(!is_dir("./upfile/")){               //判断指定目录是否存在
            mkdir("./upfile/");                 //创建目录
        }
        $path='./upfile/'.time().strstr($_FILES['up_picture']['name'],'.');//定义文件名称和存储位置
        if(is_uploaded_file($_FILES['up_picture']['tmp_name'])){//是否是通过 HTTP POST 上传的
            if(!move_uploaded_file($_FILES['up_picture']['tmp_name'],$path)){ //执行上传
                echo "上传失败";
            }else{
```



```
                    echo "文件".time().$_FILES['up_picture']['name']."上传成功，大小为：".$_FILES
['up_picture']['size'];
                }
            }else{
                echo "上传文件".$_FILES['up_pictute']['name']."不合法！";
            }
        }
    }
?>
```



专家点评

定义上传文件的名称在文件上传程序中是必不可少的一个功能，通过它可以避免重名文件被替换，但这可能会导致存在很多相同内容的文件。

问题 62　如何通过字符串截取控制图像验证码的输出?

问题阐述

对于图像验证码，相信大家并不陌生，它的主要作用是防止用户恶意刷新注册使服务器系统资源流失，数据库冗余信息堆积。图像验证码的出现避免了相关问题发生的可能性，创建 PHP 图像验证码需要开启 GD2 函数库支持，利用相关的图像函数创建。

专家解答

1. 解决思路

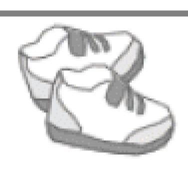
（1）定义数字验证码的取值范围，通过 rand()函数获取随机数，并且通过 substr()函数对随机数进行截取，同时将截取的值作为验证码在图像中输出。

（2）通过 GD2 函数库中的函数，生成图像验证码，其应用的 GD2 函数包括：imagecreate()、imagecolorallocate()、imagefttext()、imageline()、magesetpixel()、imagejpeg()、imagedestroy()控制并生成验证码。

2. 应用示例

本示例生成 4 位随机验证码，其关键代码如下：

```
<?php
    header("Content-Type:text/html;charset=utf-8");          //设置页面数据的编码风格
    header("Content-Type:image/jpeg");                       //告知浏览器输出的是图像
    $image=imagecreate(200,80);                              //创建画布
    $bgcolor=imagecolorallocate($image,230,230,230);         //设置背景颜色
    $fontcolor=imagecolorallocate($image,0,0,0);             //设置字体颜色
    $linecolor=imagecolorallocate($image,rand(0,200),rand(0,200),rand(0,200)); //设置线的颜色
    $pixelcolor=imagecolorallocate($image,rand(0,200),rand(0,200),rand(0,200));//设置噪点的颜色
    for($a=0;$a<10;$a++){
        $rand.=dechex(rand(0,15));                           //获取 10 个十六进制数
    }
    for($b=0;$b<4;$b++){
        $str.=substr($rand,-$b,1)."    ";                    //截取资格十六进制数
    }
```





```
        $font="STCAIYUN.TTF";                                    //定义字体
        imagefttext($image,30,rand(0,8),rand(0,40),rand(50,80),$fontcolor,$font,$str);    //输出字符
        for($c=0;$c<20;$c++){
            imageline($image,rand(0,200),rand(0,80),rand(0,200),rand(0,80),$linecolor); //输出 20 条线
        }
        for($d=0;$d<1000;$d++){
            imagesetpixel($image,rand(0,200),rand(0,80),$pixelcolor);    //输出 1000 个噪点
        }
        imagejpeg($image);                                            //输出图像
        imagedestroy($image);                                         //销毁图像
    ?>
```

运行结果如图 2.18 所示。

图 2.18　图像验证码

专家点评

使用字符串截取控制图像验证码的输出时使用了 GD2 函数库，要使用该函数库，需要修改 php.ini 文件，将“extension=php_gd2.dll”选项前面的分号去掉，然后重新启动 Apache 服务器。

问题 63　如何打乱字符串中字符的顺序？

问题阐述

打乱字符串的字符顺序，比较典型的应用是实现验证码的生成，例如，当用户登录时往往需要输入验证码，此时就需要将字符串中数字的顺序随机打乱，从而生成固定位数的验证码。

专家解答

1. 解决思路

随机打乱字符串中字符的顺序，可以使用 str_shuffle()函数。首先使用 str_shuffle()函数定义随机数，然后使用 substr()函数对随机数进行截取，最后使用 for 循环，循环输出随机数，生成数字验证码。

2. 应用示例

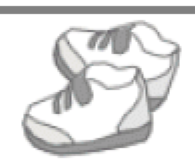
下面使用 str_shuffle()函数创建登录页面中的数字验证码，运行结果如图 2.19 所示。



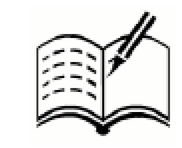
图 2.19　登录页面中的验证码

关键代码如下：

```
<?php
    $num=str_shuffle("123456789");
```



```
        $nums=substr($num,0,4);
        for($i=0;$i<4;$i++){
          echo "<img src=images/".substr(strval($num),$i,1).".gif>";
        }
    ?>
```



问题 64　如何对字符串进行大小写转换?

问题阐述

网站中有很多数据需要具有规则性，以便于管理员管理，但为了使用户输入方便，不要刻意要求用户输入大写或小写，在存储数据时可以使用程序控制以统一大写或小写的方式进行存储。

专家解答

1. 解决思路

对字符串进行大小写转换可以使用 strtoupper 函数和 strtolower 函数。strtoupper 函数可以将字符串转换为大写；strtolower 函数可以将字符串转换为小写。

2. 应用示例

下面将用户填写的图书信息中的图书编号全部以大写形式存储，运行结果如图 2.20 所示。



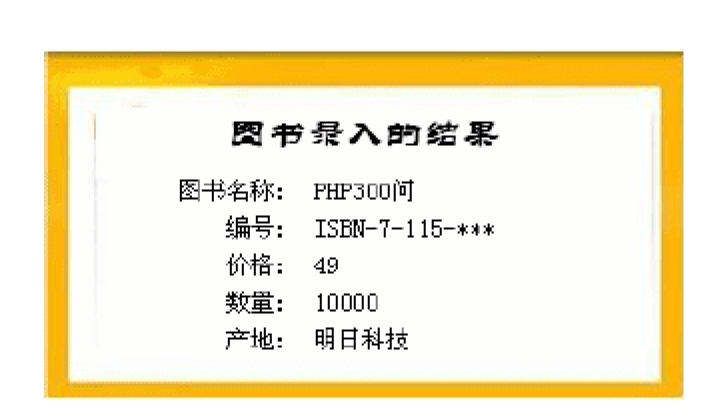


图 2.20　将图书编号转换为大写

关键代码如下：

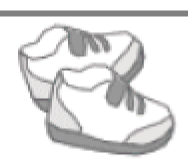
```
    <?php session_start();
        if(isset($_POST['Submit'])){
            $user=strtoupper($user);                 //将商品名称中的小写字符转换为大写
            $number=strtoupper($number);             //将编号转换为大写
          }
    ?>
```

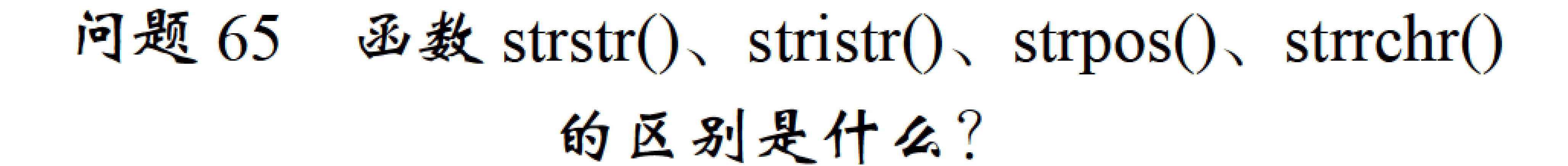
说明：

使用 strtolower 函数将字符串转换为小写的使用方法与上述方法相同。

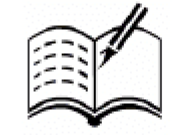
专家点评

对字符串进行大小写转换并不是必要功能，用户可以根据每个网站的不同要求选择使用。例如，在电子商务类网站的后台商品添加页面中，为了使商品的型号以统一的规格显示，经常会利用大小写转换函数，所以掌握这两个函数是很有必要的。



问题 65　函数 strstr()、stristr()、strpos()、strrchr()的区别是什么？



Note

问题阐述

strstr()、stristr()、strpos()、strrchr()函数的功能都是获取一个指定的字符串在另一个字符串中出现的位置，那么，它们之间有什么区别呢？

专家解答

1．解决思路

下面分别对这 4 个函数进行详细介绍：

（1）stristr()函数

stristr()函数用于获取指定字符串（A）在另一个字符串（B）中首次出现的位置到（B）字符串末尾的所有字符串。该函数如果执行成功则返回剩余的字符串，否则将返回 False。该函数不区分大小写，如果要在区分字母大小写的情况下进行搜索，则可以应用 strstr()函数，strstr()函数区分字母的大小写。

语法格式如下：

```
string stristr(string haystack,string needle);
```

stristr()函数的参数说明如表 2.7 所示。

表 2.7　stristr()函数的参数说明

参　　数	说　　明
haystack	必要参数，指定要查找的字符串
needle	必要参数，指定要查找的对象

（2）strstr()函数

strstr()函数用于获取一个指定字符串在另一个字符串中首次出现的位置到后者末尾的子字符串。如果执行成功，则返回剩余字符串（存在相匹配的字符）；如果没有找到相匹配的字符，则返回 False。

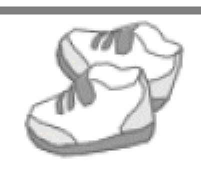
语法格式如下：

```
string strstr ( string haystack, string needle)
```

strstr()函数的参数说明如表 2.8 所示。

表 2.8　strstr()函数的参数说明

参　　数	说　　明
haystack	必要参数，指定从哪个字符串中进行搜索
needle	必要参数，指定搜索的对象。如果该参数是一个数值，那么将搜索与这个数值的ASCII值相匹配的字符





（3）strpos()函数

strpos()函数用于查找指定字符串（A）在字符串（B）中首次出现的位置。返回值为字符串所在的位置，如果失败则返回 False。本函数区分字母的大小写。

语法格式如下：

```
int strpos(string haystack,string needle,int offset)
```

strpos()函数的参数说明如表 2.9 所示。

表 2.9 strpos()函数的参数说明

参　　数	说　　明
haystack	必要参数，指定从哪个字符串中开始检索字符串
needle	必要参数，指定要检索的字符串
offset	必要参数，指定从哪个字符串开始检索

（4）strrchr()函数

strrchr()函数用于查找指定字符串（A）在另一个字符串（B）中最后一次出现的位置。返回从字符串（B）中的这个位置起，一直到字符串（B）结束的所有字符；如果失败则返回 false。本函数区分字母的大小写。

语法格式如下：

```
string strrchr(string haystack,string needle)
```

strrchr()函数的参数说明如表 2.10 所示。

表 2.10 strrchr()函数的参数说明

参　　数	说　　明
haystack	指定从哪个字符串中进行搜索
needle	指定要搜索的字符串，如果要搜索的是一个数字，那么将搜索与该数字对应的ASCII值相匹配的字符

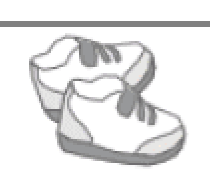
2. 应用示例

下面分别用 strstr()、stristr()、strpos()和 strrchr()函数查找同样的字符串，观察其输出结果，代码如下：

```
<?php
$str = 'mingrisoft@Mingrisoft.com';
echo strstr($str, 'M').'<br/>';    //提取从查找字符串第一次出现到最后的字符串，区分大小写
echo stristr($str, 'M').'<br/>';   //提取从查找字符串第一次出现到最后的字符串，不区分大小写
echo strpos($str,'M').'<br/>';     //返回字符串第一次出现位置，区分大小写
echo strrchr($str, 'M').'<br/>';   //提取从查找字符串最后一次出现到最后的字符串，区分大小写
?>
```

运行结果为：

```
Mingrisoft.com
mingrisoft@Mingrisoft.com
11
Mingrisoft.com
```





专家点评

上述 4 个函数虽然相似，但都有各自不同的用处，读者一定要分清每个函数实现的功能。

问题 66　if 和 switch 语句有何区别？

问题阐述

if 和 switch 语句有何区别？

专家解答

if 和 switch 语句可以从使用的效率上来进行区分，也可以从实用性角度去区分。如果从使用的效率上进行区分，在对同一个变量的不同值作条件判断时，使用 switch 语句的效率相对更高一些，尤其是判断的分支越多越明显。

如果从语句的实用性的角度去区分，那么 switch 语句肯定不如 if 条件语句，可以说，if 条件语句是实用性最强和应用范围最广的语句。

在程序开发的过程中，if 和 switch 语句的使用应该根据实际的情况而定，不要因为 switch 语句的效率高就一味地使用，也不要因为 if 语句常用就不应用 switch 语句。用户应根据实际的情况，具体问题具体分析，使用最适合的条件语句。一般情况下可以使用 if 条件语句，但是在实现一些多条件的判断中，特别是在实现框架的功能时就应该使用 switch 语句。

问题 67　如何防止页面出现中文乱码？

问题阐述

如何防止页面出现中文乱码？

专家解答

在编写 PHP 程序时，为了防止页面出现乱码，要求 PHP 文件编码、网页页面编码和数据库编码一致，如果采用 UTF-8 编码，则上述 3 项内容都要设置为 UTF-8 编码，否则会出现乱码现象。那么如何对上述内容编码进行设置呢？设置 PHP 文件编码可以采用 UltraEdit、Edit Plus 和 Zend Studio 等编辑器，如果使用 UltraEdit 设置 PHP 页面文件编码，则可以在另存页面文件时选择格式为 UTF-8，如图 2.21 所示。

设置网页的页面编码可以使用 PHP 中的 header()函数或在 HTML 的 Meta 标记中设置。如果页面完全由 PHP 脚本编写，则可以使用 header()函数设置页面编码。设置方法如下所示：

```
header('content-type:text/html; charset=UTF-8');
```

如果页面中包含规范的 HTML 代码，则可以在 HTML 的 Meta 标记中设置页面编码，其设置方法如下所示：

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8" />
```



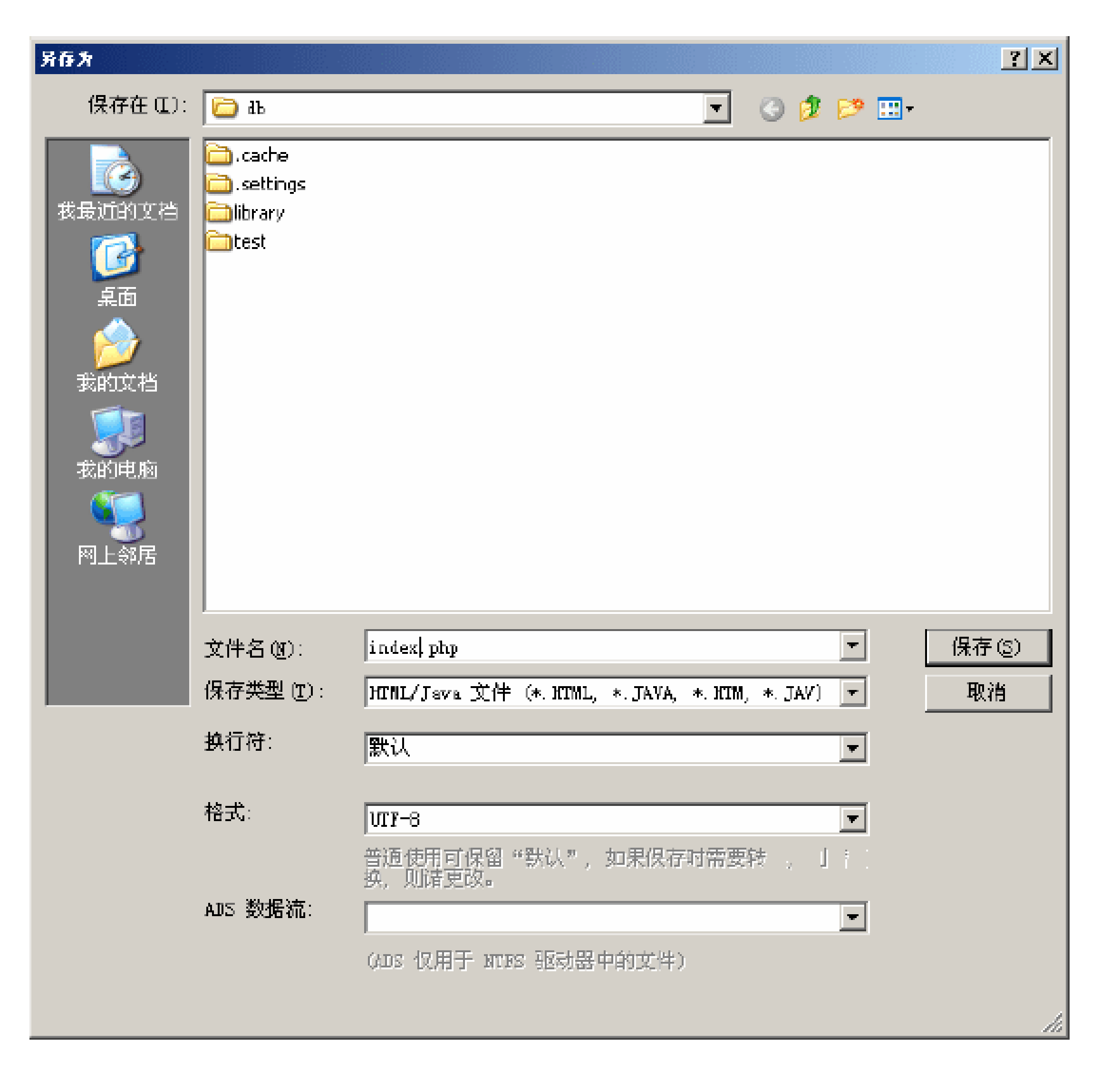




图 2.21　使用 UltraEdit 另存 PHP 文件为 UTF-8 编码

完成对 PHP 文件及页面编码的设置后，还需要设置数据库编码，一般需要进行两处设置，在创建数据表时要设置字段的字符编码，其设置方法如图 2.22 所示。

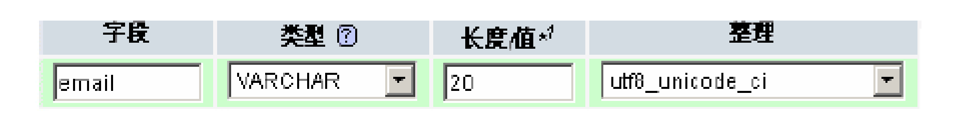


图 2.22　设置表中字段的字符编码

完成上述设置后，一般还需要在程序中采用编码方式设置数据库字符集，其设置方法可以通过 SQL 语句实现，例如，通过如下语句设置数据库字符集为 UTF-8。

```
$conn->query("set names utf8");
```

专家点评

在设置数据库字符编码为 UTF-8 时，应为 set names utf8 而非 set names utf-8，注意不要多加字符“-”。

第 3 章

Web 技术

- ⏭ 问题 68　如何打印指定框架中的内容?
- ⏭ 问题 69　如何使用框架嵌套技术居中显示框架页?
- ⏭ 问题 70　如何为文本框设置只读属性?
- ⏭ 问题 71　如何实现自动计算金额?
- ⏭ 问题 72　如何实现复选框中的全选、反选和不选?
- ⏭ 问题 73　如何对上传的图片进行预览?
- ⏭ 问题 74　如何制作在页面右下角弹出渐显的广告窗口?
- ⏭ 问题 75　判断表单动作的技巧有哪些?
- ⏭ 问题 76　如何使用 JS 脚本获取、输出标签内容?
- ⏭ 问题 77　如何控制客户端 COOKIE?
- ⏭ 问题 78　COOKIE 应用的注意事项有哪些?
- ⏭ 问题 79　SESSION 和 COOKIE 的区别有哪些?
- ⏭ 问题 80　如何使用 SESSION?
- ⏭ 问题 81　如何结束所有 SESSION?
- ⏭ 问题 82　如何阻止浏览器缓存页面?
- ⏭ 问题 83　如何控制页面背景的打印?
- ⏭ 问题 84　如何实现文字滑动效果?
- ⏭ 问题 85　如何实现自动登录?
- ⏭ 问题 86　如何统计用户在线时间?
- ⏭ 问题 87　如何限制用户访问网站的时间?
- ⏭ 问题 88　如何掌控登录用户的权限?
- ⏭ 问题 89　如何让 SESSION 永久有效?
- ⏭ 问题 90　SESSION 购物车
- ⏭ 问题 91　header()函数都有哪些功能?





问题 68　如何打印指定框架中的内容？

问题阐述

在网页中实现打印功能时，打印的是整个网页中的所有内容，但很多时候我们想要打印的只是整个网页中的部分内容，该如何实现呢？这就是我们在这里要介绍的方法——使用框架实现网页中部分内容的打印。

专家解答

1. 解决思路

要打印页面中指定的内容，可以利用框架中各部分相互独立的特点，将要打印的内容放置在一个框架中，在打印该内容时就不会牵涉到整个页面的内容了。其具体方法如下：

```
parent.mainFrame.fcous();
window.print();
```

mainFrame：表示框架的名称。

通过 parent.mainFrame.fcous()方法获取框架焦点，然后应用 window 对象的 print()方法实现打印功能。

2. 应用示例

运行 frame 框架实现页面内容的部分打印。其具体步骤如下：

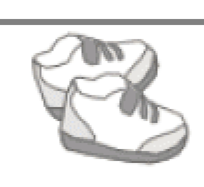
（1）首先设计 content.php 页面，应用 include 调用数据源文件，再应用 do…while 循环语句将要打印的内容显示在该页面中。

```
<?php
include "conn/conn.php";
      $sql=mysql_query("select * from tb_user");
      $info=mysql_fetch_array($sql);
do{?>
<tr>
   <td height="30"><?php echo $info[username];?></td>
   <td><?php echo $info[tel];?></td>
   <td><?php echo $info[linker];?></td>
   <td><?php echo $info[linktel];?></td>
   <td><?php echo $info[email];?></td>
   <td><?php echo $info[address];?></td>
  </tr>
<?php }while($info=mysql_fetch_array($sql))?>
```

（2）设计 index.php 页面，在该页面的适当位置添加浮动框架（应用<iframe>标记创建），命名为“content”，并将该浮动框架的 src 属性指定为步骤（1）中创建的 content.php 文件，代码如下：

```
<iframe name="content" src="content.php"   frameborder="0" width="100%" height="100%"></iframe>
```

（3）在 index.php 页面中添加“打印”超链接，打印指定浮动框架 content 中的内容，代





码如下：

```
<a href="#" onClick="parent.content.focus();window.print();">打印</a>
```

其运行结果如图 3.1 所示，单击“打印”超链接，弹出打印对话框，进行相应的设置后，即可打印文本框中的内容。



图 3.1　打印指定框架中的内容

问题 69　如何使用框架嵌套技术居中显示框架页？

问题阐述

使用框架可以将一个完整的页面分割成不同的窗口，每个窗口相互独立，且每个窗口中的内容在运行时互不干扰。虽然将页面分割成多个窗口在某些时候较为实用，但也存在一些问题，就是如何能够让分割后的整个页面居中显示，因为我们所使用的计算机显示器的大小不同，在输出内容时会随着显示器大小的变化而变化，为了让网页更加美观，必须保证其居中输出。那么如何才能保证框架中的内容居中显示呢？

专家解答

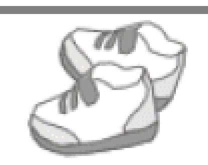
框架是指网页的各部分是相互独立的页面，并且通过一个网页将这些相互独立的网页组成一个完整的网页显示在浏览器中，重复出现的内容被固定下来，当每次浏览者发出对网页的请求时，只下载发生变化的框架页面，其他的页面保持不变。使用框架可以将浏览器的窗口划分为若干个子窗口，在每个子窗口可以分别显示不同的网页。

1. 框架网页的基本结构

框架网页的基本结构如下：

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
<title>框架网页</title>
```







```
</head>
<frameset>
   <frame>
   <frame>
</frameset>
<noframes>
<body>
对不起!您的浏览器不支持框架网页的显示!
</body>
</noframes>
</html>
```

2. 设置框架集的属性

框架集包含如何组织各个框架的信息，可以通过<frameset>标记定义。框架是按照行和列来组织的，可以使用<frameset>标记的属性对框架的结构进行设置。

（1）左右分割窗口属性 cols，在水平方向上将浏览器分割成多个窗口，其语法格式如下：

```
<frameset cols="value,value,......">
   <frame>
   <frame>
</frameset>
```

value：用于指定各个框架的列宽，取值有 3 种形式：像素、百分比（%）和相对尺寸（*）。

说明：

如果将 cols 属性设置为 “*，*，*”，则表示将窗口划分成 3 个等宽的框架；如果将 cols 属性设置为 “*，2*，3*”，则表示左边的框架占窗口宽度的 1/6，中间的框架占窗口宽度的 1/3，右边的框架占窗口宽度的 1/2。

（2）上下分割窗口属性 rows，在垂直方向上将浏览器分割成多个窗口，其语法格式如下：

```
<frameset rows="value,value,......">
   <frame>
   <frame>
</frameset>
```

value：用于指定框架的行高，取值有 3 种形式：像素、百分比（%）和相对尺寸（*），设置的方式与 cols 属性类似。

（3）框架边框显示属性 frameborder

该属性用于指定框架周围是否显示边框，取值有 1（显示边框，默认值）和 0（不显示边框）。

（4）framespacing

该属性用于指定框架之间的间隔，以像素为单位。如果不设置该属性，则框架之间没有间隔。

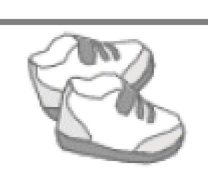
（5）指定边框宽度属性 border

该属性用于指定边框的宽度，只有当 frameborder 属性为 1 时有效。

3. 设置框架的属性

使用<frame>标记可以设置框架的属性，包括框架的名称、是否包含滚动条以及在框架中显示网页等。其语法格式如下：

```
<frame src="文件" name="名称" scrolling="值" [noresize] frameborder="数值">
```





Note

参数说明：

❶ name：指定框架名称。

❷ src：指定在框架中显示的网页文件。

❸ frameboder：指定框架周围是否显示边框，取值有 1（显示边框）和 0（不显示边框）。

❹ noresize：可选属性，如果指定了该属性，则不能调整框架的大小。

❺ scrolling：指定框架是否包含滚动条。如果将该属性设置为“yes”，则框架包含滚动条；如果将该属性设置为“no”，则表示不包含滚动条；如果将该属性设置为“auto”，则在需要时包含滚动条。

4．应用示例

在使用框架技术设计网站布局时，有一个问题是必须要考虑的，那就是浏览器的分辨率。如果使用 1024*768 的分辨率制作框架，那么页面在更高分辨率的显示器中将会是什么样子呢？这就需要在使用框架设计布局时增加一个框架嵌套技术，使框架页在浏览器中居中显示。其具体方法如下：

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
<title>使用框架嵌套技术居中显示框架页</title>
</head>
<frameset rows="*" cols="*,630,*" frameborder="no" framespacing="0" border="0">
    <frame src="blank.php" name="BLFrame" scrolling="no" noresize>
        <frameset rows="55,*" cols="*" frameborder="no" border="0" framespacing="0">
     <frame src="software_top.php" name="topFrame" scrolling="no" noresize>
     <frameset rows="*" cols="160,*" framespacing="0">
        <frame src="software_left.php" name="leftFrame" scrolling="auto" noresize>
        <frame src="software_main.php" name="mainFrame" frameborder="no" scrolling="auto">
     </frameset>
        </frameset>
     <frame src="blank.php" name="BRFrame" scrolling="no" noresize>
</frameset>
<noframes>
<body>
该浏览器不支持框架，无法正常显示!!
</body>
</noframes>
</html>
```

在此框架中包括 4 个子页面路径：blank.php、software_top.php、software_left.php 和 software_main.php。其运行结果如图 3.2 所示。无论在多大分辨率的显示器下，其都能够正常地居中显示。

专家点评

这里介绍的方法不仅应用了框架技术，同时还应用了框架嵌套技术，使网页无论在多大分辨率的显示器中都能够居中显示。框架嵌套技术主要就是在划分浏览器的窗口时首先对窗口进行一次左右的分割，cols 的值“cols="*,630,*"”，它表示该框架的中间窗口为 630 像素，且左右窗口平分。其中中间窗口的宽度就是网页界面的实际宽度，然后，在此基础上再进行其他窗口的划分。

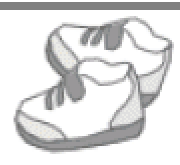
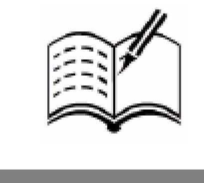




图 3.2　使用框架嵌套技术居中显示框架页



问题 70　如何为文本框设置只读属性？

问题阐述

在网页中，有些文本框中的信息是不允许浏览者进行修改的，即用户只有浏览的权限，而没有修改的权限，如何实现此功能呢？

专家解答

对文本框设置只读属性主要是对文本框中 readonly 属性的运用，通过它设置文本框的只读属性，然后通过 onfocus 事件调用 JavaScript 脚本，当用户单击已经设置为只读属性的文本框时，弹出一个提示对话框。

使用 readonly 属性实现只读的代码如下：

```
<input id="o" class="o" type="text" readonly value='0100521041'/>
```

例如，设置表单中输出的库存数量文本框为只读属性，并且通过 value 属性设置文本框的默认值。其代码如下：

```
<form action="" method="post">
商品编号：<input id="o" class="o" type="text" readonly value='0100521041'/><br>
商品名称：<input id="o"type="text" value="酸牛奶" onfocus="this.value="" /><br>
库存数量：<input id="o" class="n"type="text" disabled="disabled" value="100 件" /><br>
管理员工：<input id="o"type="text" value="001" />
</form>
```

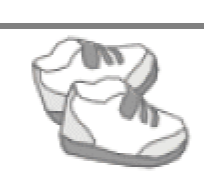
专家点评

在使用 readonly 时，只需在<input>标记中加入属性名称即可。

问题 71　如何实现自动计算金额？

问题阐述

在物流管理系统中，经常需要计算出库商品的金额或入库商品的金额，若以手工计算比较



麻烦而且容易出错，如果在程序中实现自动计算金额功能，则可以减少许多不必要的麻烦。

专家解答



要实现自动计算金额，主要通过触发表单控件的 Blur 事件，即表单控件失去焦点时，实现自动计算金额。实现此功能的 JavaScript 代码如下：

```
$(".o").blur(function(){                    //当商品单价文本框失去焦点
  var price = $(".o").val();
  var num = $(".n").val();
  var amount = price * num;
  $(".e").val(amount);
});
```

例如，通过 jQuery 脚本自动计算金额，并设置输出金额的文本框为只读，其代码如下：

```
<form action="" method="post" name="form1">
商品编号：<input id="o" type="text" readonly value='酸牛奶'/><br>
商品单价：<input id="o" type="text"class="o" value="0.0"/><br>
库存数量：<input id="o" class="n"type="text" value="35" readonly /><br>
金额总计：<input id="o" class="e"type="text" value="0.0" />
</form>
```

运行结果如图 3.3 所示。



图 3.3　自动计算金额

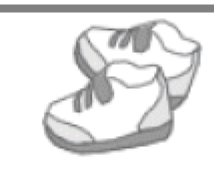
专家点评

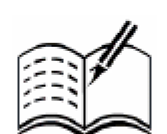
用于当 Blur 事件（即失去焦点事件）发生时执行 JavaScript 代码。适用对象包括 Button、Checkbox、FileUpload、Layer、Password、Radio、Reset、Select、Submit、Text、Textarea 和 Window。

问题 72　如何实现复选框中的全选、反选和不选？

问题阐述

复选框常被应用在数据管理系统中，如多数据的删除等。如果想要删除全部的数据，手动选择每条数据前面的复选框，显得过于麻烦，那么该如何实现复选框的全选、反选和不选呢？





专家解答

1. 解决思路

复选框的全选、反选和不选主要应用 JavaScript 脚本，通过自定义函数完成。

（1）在通过 JavaScript 脚本中的自定义函数完成复选框的全选、反选和不选功能中，应用的第一个方法是：getElementsByTagName，获取指定标签的名称，返回值为一个包含标签信息的 object。

（2）根据 getElementsByTagName 标签返回的对象，判断标签类型（type）的值是否是 checkbox。

（3）当标签类型 type 的值为 checkbox 时，为标签中的 checked 赋值。当 checked 的值为 true 时，为它赋值为 false；当 checked 的值为 false 时，为它赋值为 true。

（4）在页面中通过 script 标签调用 js 文件夹下的文件，应用 onClick 事件调用自定义函数，实现全选、反选和不选功能。

2. 应用示例

在本示例中实现复选框的全选、反选和不选功能。其具体步骤如下：

（1）创建 js 文件夹，编写 reg.js 脚本文件。然后在 reg.js 中编写自定义函数，实现全选、反选和不选功能。reg.js 文件的关键代码如下：

```
function uncheckAll(form1,status) {                             //不选
    var elements = form1.getElementsByTagName('input');         //获取 input 标签
    for(var i=0; i<elements.length; i++){                       //根据标签的长度执行循环
        if(elements[i].type == 'checkbox') {                    //判断对象中元素的类型
            if(elements[i].checked==true){                      //判断当 checked 的值为 true 时
                elements[i].checked=false;                      //为 checked 赋值为 false
            }
        }
    }
}
function checkAll(form1,status){                                //全选
    var elements = form1.getElementsByTagName('input');
    for(var i=0; i<elements.length; i++){
        if(elements[i].type == 'checkbox') {
            if(elements[i].checked==false){
                elements[i].checked=true;
            }
        }
    }
}
function switchAll(form1,status) {                              //反选
    var elements = form1.getElementsByTagName('input');
    for(var i=0; i<elements.length; i++){
        if(elements[i].type == 'checkbox'){
              if(elements[i].checked==true){
                elements[i].checked=false;
              }else if(elements[i].checked==false){
                elements[i].checked=true;
            }
        }
```



```
        }
    }
```

（2）创建 index.php 页面，输出会员信息，并添加图像按钮，通过 onClick 事件调用 JavaScript 自定义函数实现全选、反选、不选和删除的功能。其关键代码如下：

```
<script language="javascript" src="js/reg.js"></script>
<form method="post" name="form1" id="form1" action="index_ok.php">
<tr>
<td width="62"align="center"><input name="conn_id[]" type="checkbox" id="conn_id[]"value="1"/></td>
    <td width="242"><?php echo $myrow[id];?></td>
    <td width="243"><?php echo $myrow[user];?></td>
    <td width="485"><?php echo $myrow[pass];?></td>
    <td width="485"><?php echo $myrow[dates];?></td>
</tr>
<tr>
<td colspan="5" align="center"><!--通过 onClick 事件调用自定义函数，执行相应的操作-->
<img src="images/bg_19-20.jpg" onclick="checkAll(form1,status)" width="62" height="25" />
    <img src="images/bg_14-14.jpg" onclick="switchAll(form1,status)"   width="62" height="25" />
    <img src="images/bg_07-08.jpg" width="62" height="25" onclick="uncheckAll(form1,status)" />
<input type="image" name="imageField" src="images/bg_14.jpg" />
</td>
</tr>
</form>
```

专家点评

复选框的全选、反选和不选主要应用 JavaScript 脚本，通过 JavaScript 脚本中的自定义函数，判断复选框中 checked 的值。全选就是为 checked 赋值为 true，不选则是为 checked 赋值为 false，而反选则是当 checked 的值为 true 时，为它赋值为 false；当 checked 的值为 false 时，为它赋值为 true。

问题 73　如何对上传的图片进行预览？

问题阐述

在上传图片时，往往会因为没有事先查看图片而传错图片，如果可以实现上传图片的预览功能，就不会出现传错图片的问题了。那么如何实现上传图片的预览功能呢？

专家解答

实现上传图片的预览功能应用的是 jQuery 框架中的方法，直接读取上传图片，并且将读取的结果输出，同时还判断上传图片是否为空。其方法存储于 in.js 脚本文件中，代码如下：

```
$(document).ready(function(){
    $("#two").click(function(){                          //单击预览图片
        var value = $("#one").val();                     //取得上传地址
        if(value == ""){                                 //判断地址是否为空
            alert("地址为空");
        }else{
```






```
                    $(".n").append("<img id='five' src="+value+">");          //显示图片
                    $("#two").click(function(){
                         $(".n").empty();
                         $(".n").append("<img id='five' src="+value+">");
                    });
               }
          });
          $("#three").click(function(){                                        //显示图片地址
               var value = $("#one").val();
               if(value == ""){
                    alert("地址为空");
               }else{
                    alert(value);
               }
          });
          $("#four").click(function(){                                         //清除文本框信息
               $(".e").empty();
               $(".e").append("<input class='one' id='one' type='file'>");
          });
     });
```

创建 index.php 文件，引入 CSS 样式和 in.js 文件以及 jQuery 库文件 jquery-1.3.2.js。创建 form 表单，设置文件域属性，添加预览图片、图片地址和清除的按钮。实现图片上传和预览的功能，其代码如下：

```
<table width="600" height="450" align="center" background="pic/bg.jpg">
   <tr>
     <td align="center"><div class="n"></div>
<table align="center">
   <tr>
<td>
          <span class="o">
          <span class="e"><input class="one" id="one" type="file"></span>
          <input class="one" id="two" type=button value="预览图片">
          <input class="one" id="three" type=button value="图片地址">
          <input class="one" id="four" type=button value="清除" >
          </span>
</td>
   </tr>
</table>
     </td>
   </tr>
</table>
```

运行结果如图 3.4 所示。

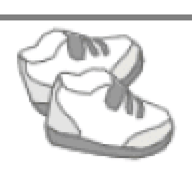
专家点评

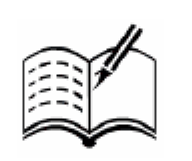
jQuery 框架中的 empty()方法删除匹配的元素集合中所有的子节点。其语法如下：

```
$('class/id').empty();
```

参数说明：

class/id：表示类名或 id 名称。





Note



图 3.4　上传预览图片

问题 74　如何制作在页面右下角弹出渐显的广告窗口？

问题阐述

广告作为网站最大的盈利手段，任何网站都不可能将其省略，那么，如何才能实现网站既可以从广告中获利，又能使用户流畅地阅读网站信息呢？

专家解答

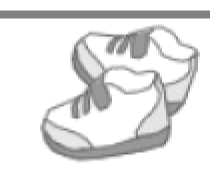
从首页右下角弹出广告窗体应用的是 JavaScript 脚本，通过 window 对象在 HTML 中重新打开一个窗口，应用 open 方法控制打开的文件及窗口的大小，通过 screen 方法控制窗口在页面中的位置。其具体方法的代码如下：

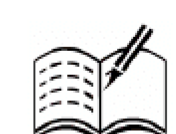
```
<script language="javascript">
var newformW=300;
var newformH=180;
function pp(){
    var T=window.screen.width-newformW;
    var L=window.screen.height+newformH;
    var  name=window.open("advertise.htm","","width="+newformW+",height="+newformH+",top="+T+",left="+L);
}
pp();
</script>
```

运行结果如图 3.5 所示。



图 3.5　右下角渐显广告





专家点评

window 对象打开窗口详解。

使用 window 对象打开窗口的语法格式如下：

```
windowVar=window.open(url,windowname[,location]);
```

参数说明：

❶ windowVar:：当前打开窗口的句柄。如果 open()方法成功，则 windowVar 的值为一个 window 对象的句柄，否则 windowVar 的值是一个空值。

❷ url：目标窗口的 URL。如果 URL 是一个空字符串，则浏览器将打开一个空白窗口，同时允许用 write()方法创建动态 HTML。

❸ windowname：window 对象的名称。

❹ location：对窗口属性进行设置。

问题 75　判断表单动作的技巧有哪些？

问题阐述

表单可以通过同一个程序来分配应该要处理的动作，在表单中有不同的逻辑，应如何判别使用者按下的按钮呢？

专家解答

要解决这个问题很简单，只要通过提交按钮的 name 就可以知道了，表单在提交出去时，只有按下的 submit 类型的按钮才会被送到表单数组去，所以只要判断按钮的值就可以知道使用者按下哪一个按钮，以如下表单为例：

```
<form id="form1" name="form1" method="post" action="">
<input type="submit" name="btn" value="a" />
<input type="submit" name="btn" value="b" />
</form>
```

当使用者按下“a”按钮时，btn=a；按下“b”按钮时，btn=b。

另外，也可以通过提交按钮的名字（name）进行判断，见如下代码：

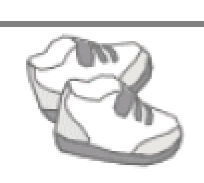
```
<form id="form1" name="form1" method="post" action="">
<input type="submit" name="a" value="提交 A" />
<input type="submit" name="b" value="提交 B" />
</form>
```

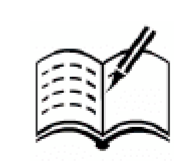
这样，只要 POST/GET 的参数中有 a 或 b，就可以知道按下的是哪个按钮了。

问题 76　如何使用 JS 脚本获取、输出标签内容？

问题阐述

如何获取、更改表单元素值和特定标签内容？





Note

专家解答

使用 JS 脚本获取页面内容的方式主要有两种，第一种方式是通过表单获取表单元素的 value 值。格式为：



```
表单名称.元素名.value;
```

该方式只能获取表单中的元素值，对于其他标签元素则无能为力。

第二种方式是通过 id 名来获取页面中任意标签的内容。格式为：

```
document.getElementById('id'). value;
document.getElementById ('id').innerText;
```

使用第二种方式时要注意，标签的 id 名必须存在且唯一，否则就会出现错误。

当为标签内容赋值时，使用如下格式：

```
id.innerHTML ='要显示的内容';
```

问题 77　如何控制客户端 COOKIE?

问题阐述

如何控制客户端 COOKIE?

专家解答

对 COOKIE 的控制属于 IE 浏览器 Internet 属性中隐私的内容。操作方法如下：

（1）选择 IE 浏览器，单击鼠标右键，在弹出的快捷菜单中选择“属性”命令，在弹出的 Internet 属性对话框中选择“隐私”选项卡，如图 3.6 所示。

（2）移动图 3.6 中的滑块，设置 Internet 区域的隐私。

（3）设置完成后，单击“应用”按钮，保存设置。



图 3.6　控制客户端 COOKIE

问题 78　COOKIE 应用的注意事项有哪些?

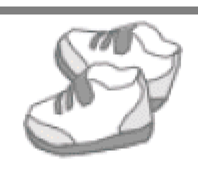
问题阐述

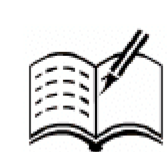
在学习及使用 COOKIE 时，都应该注意哪些事项呢?

专家解答

1. 牢记 setcookie()函数的特性

在使用 setcookie()函数创建 COOKIE 之前，不能有任何 HTML（即使是空格、空行也不可以）输出，这点与 session_start ()函数类似。如果犯了这个错误，那么得到的将是一堆错误代码。





2. 为什么$_COOKIE 在当前页获取不到 COOKIE 值

通过 setcookie()函数创建 COOKIE 后，在当前页应用$_COOKIE 获取不到 COOKIE 的值，而必须是在刷新或到达下一页面后才能获取到 COOKIE 值。因为 setcookie()函数执行后，向客户端发送一个 COOKIE，如果不刷新或浏览下一个页面，客户端就不能将 COOKIE 送回。

3. COOKIE 会一直伴随网站中的每个请求，直到 COOKIE 过期

浏览器创建一个 COOKIE 后，对于每一个针对该网站的请求，都会在 Header 中带着这个 COOKIE，而且浏览器会这样一直发送，直到 COOKIE 过期为止。不过，对于其他网站的请求 COOKIE 是绝对不会跟着发送的。

4. 使用 COOKIE 的限制

（1）如果 COOKIE 不设置过期时间，那么它的生命周期就是浏览器会话的期间，只要关闭浏览器，COOKIE 就消失，这是会话 COOKIE，它一般不保存在硬盘上，而是保存在内存中。

（2）如果 COOKIE 设置过期时间，那么浏览器就会把 COOKIE 保存到硬盘中，再次打开浏览器时依然有效，直到它过期为止。

（3）不是所有的浏览器都支持 COOKIE。

（4）COOKIE 数据是以明文的形式存储于客户端的计算机中，不适合保存敏感的、未加密的数据。

（5）COOKIE 存储数据量有一定的限制。一个浏览器允许最多存储 300 个 COOKIE 文件，而且每个 COOKIE 文件支持最大容量为 4KB；每个域名最多支持 20 个 COOKIE，如果达到限制，浏览器会自动随机删除 COOKIE。

问题 79　SESSION 和 COOKIE 的区别有哪些？

问题阐述

SESSION 和 COOKIE 的区别有哪些？

专家解答

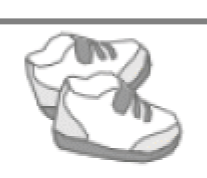
SESSION 和 COOKIE 最大的区别是：

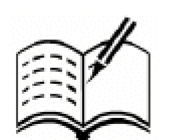
SESSION 将信息保存在服务器上，并通过一个 SESSION_ID 来传递客户端的信息。服务器在接收到 SESSION_ID 后根据这个 ID 来提供相关的 SESSION 数据。COOKIE 是将所有的信息以文本文件的形式保存在客户端，并由浏览器进行管理和维护。

因为 SESSION 数据存储于服务器端，所以远程用户无法对 SESSION 数据进行修改。而 COOKIE 数据存储于客户端，很容易被获取和篡改。所以 SESSION 的安全性要比 COOKIE 高。

专家点评

对于 SESSION 和 COOKIE 的选择，只有从实际的需求出发，选择最适合的技术才能够开发出最理想的 Web 程序。





Note

问题 80　如何使用 SESSION?

问题阐述

在 Web 程序中，如何使用 SESSION 呢？

专家解答

SESSION 被译成中文为“会话”，其本义是指有始有终的一系列动作/消息，如打电话时从拿起电话拨号到挂断电话这中间的一系列过程可以称之为一个 SESSION。

在计算机专业术语中，SESSION 是指一个终端用户与交互系统进行通信的时间间隔，通常指从注册进入系统到注销退出系统之间所经过的时间。其工作原理如图 3.7 所示。

如图 3.7 所示，当登录网站时，启动 SESSION 会话，在服务器中随机生成一个唯一的 SESSION_ID，这个 SESSION_ID 在本次登录结束之前在页面中一直有效。当关闭页面或执行注销操作后，这个 SESSION_ID 会在服务器中自动注销。当重新登录此页面时，会再次生成一个随机且唯一的 SESSION_ID。

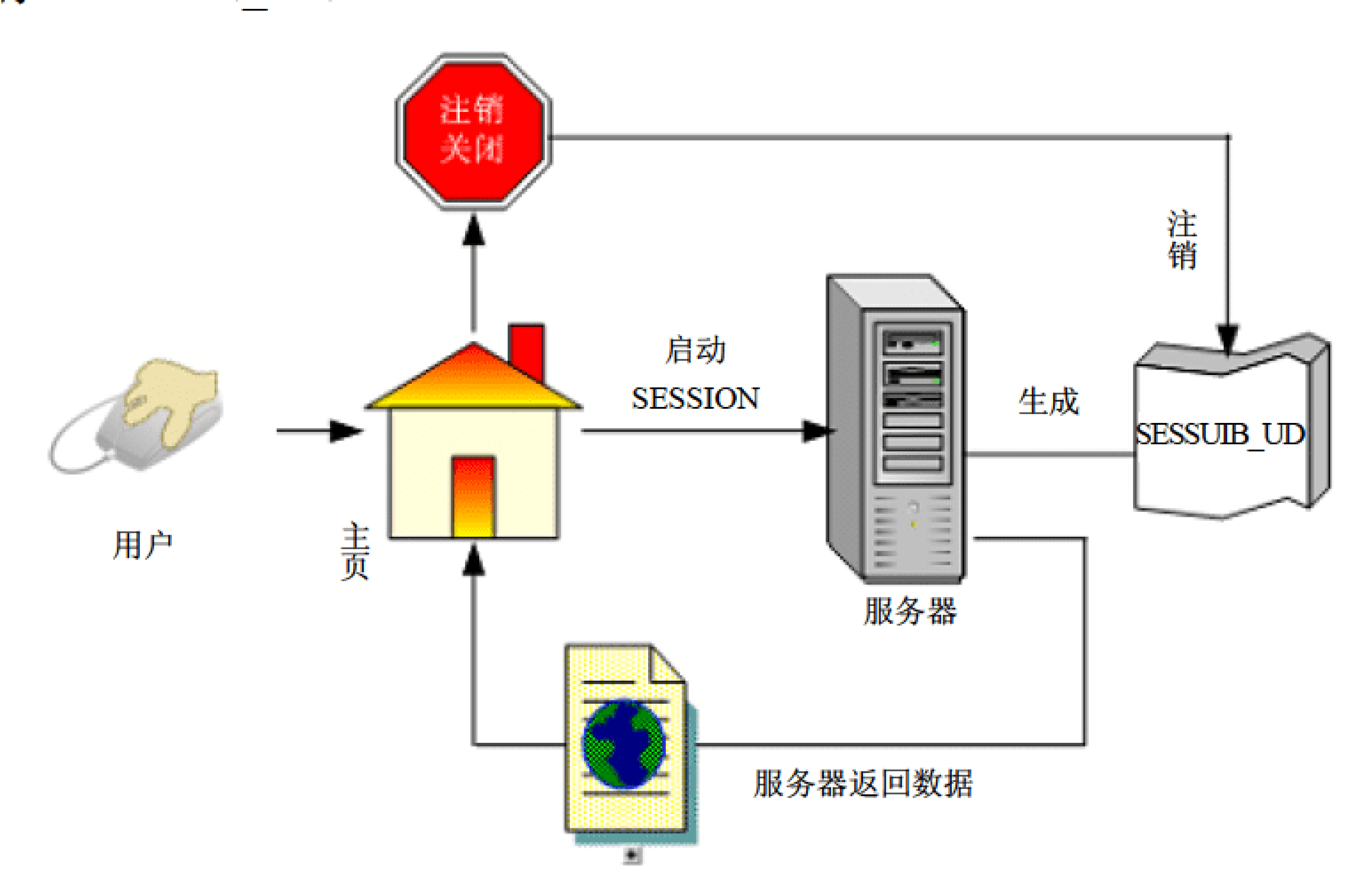


图 3.7　SESSION 工作原理

（1）使用 SESSION 前，必须先调用 session_start()函数，该函数主要用于初始化一个新的 SESSION，语法格式如下：

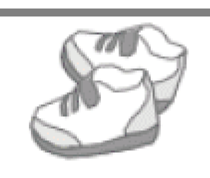
```
boolean session_start(void)
```

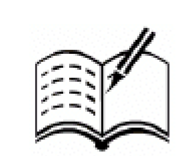
session_start()函数的参数为空。

（2）会话变量被启动后，全部保存在数组$_SESSION 中。通过数组$_SESSION 创建一个会话变量很容易，只要直接给该数组添加一个元素即可。

例如，启动会话，创建一个 SESSION 变量并赋予空值，代码如下：

```
<?php
session_start();                          //启动 SESSION
$_SESSION["user"] = null;                 //声明一个名为 user 的变量，并赋空值
?>
```





专家点评

SESSION 在 Web 技术中占有非常重要的地位。由于网页是一种无状态的连接程序，所以无法获取用户的浏览状态。因此必须通过 SESSION 记录用户的有关信息，以供用户再次以此身份对 Web 服务器提供要求时做确认。例如，在电子商务网站中，通过 SESSION 记录用户登录的信息，判断用户的访问权限，通过 SESSION 实现购物车功能，记录用户所购买商品。

问题 81　如何结束所有 SESSION?

问题阐述

SESSION 使用完成后，如何结束 SESSION 呢？

专家解答

删除会话有 3 种方式：删除单个会话、删除多个会话和结束当前会话。

☑ 删除单个会话

删除会话变量与删除数组中的指定元素相同，都可以通过 unset()函数完成。语法如下：

```
unset ( $_SESSION['user'] ) ;
```

其参数是$_SESSION 数组中的指定元素，该参数不可以省略。通过 unset()函数一次删除数组中的一个元素；如果通过 unset()函数一次注销整个数组（unset($_SESSION)），那么会禁止整个会话功能，而且无法将其恢复，用户也不能再注册$_SESSION 变量。所以，如果读者要删除多个或全部会话，则可以采用下面的两种方式。

☑ 删除多个会话

如果要一次注销所有的会话变量，则可以将一个空的数组赋值给$_SESSION，代码如下：

```
$_SESSION = array() ;
```

☑ 结束当前会话

如果整个会话已经结束，那么就可以使用 session_destroy()函数结束当前的会话。语法如下：

```
session_destroy();
```

该函数清空会话中的所有资源，彻底销毁 SESSION。

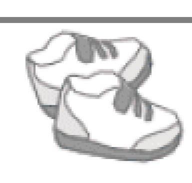
问题 82　如何阻止浏览器缓存页面?

问题阐述

如果“信息及时”对站点至关重要，并且希望过时的信息不会再被看到，就需要知道如何从一开始就阻止浏览器和代理服务器缓存页面？

专家解答

解决这个问题有两个可用的方法，即使用 HTML meta 标签和使用 HTTP 头。





1. 使用 HTML meta 标签

阻止缓存页面最常用的方法是使用 HTML meta 标签：

```
<meta http-equiv="expires" content="Wed,13 Jul 2011 05:00:00 GMT"/>
<meta http-equiv="pragma" content="no-cache"/>
```

已经传递到 Expires meta 标签的插入日期告诉浏览器，缓存的页面拷贝总是过期的。遇到此标签后，浏览器通常不会缓存页面。虽然 Pragma: no-cache meta 标签并不是保证，但它却是大多数网络浏览器遵从的受到良好支持的约定。

2. 使用 HTTP 头

阻止缓存页面更好的办法是应用 header()函数，使用 HTTP 本身协议生成与上面等价的两个 HTML meta 标签：

```
<?php
header('Expires:Wed,13 Jul 2011 05:00:00 GMT');
header('Pragma:no-cache');
?>
```

可更进一步使用 HTTP1.1 的浏览器所支持的 Cache-Control 头：

```
<?php
header('Expires:Wed,13 Jul 2011 05:00:00 GMT');
header('Cache-Control: no-store,no-cache,must-revalidate');
header('Cache-Control: post-check=0,pre-check=0',FALSE);
header('Pragma:no-cache');
?>
```

专家点评

虽然使用 Expires meta 标签是一个很好的办法，但是随之而来的有两个问题：

第一，浏览器必须下载页面以读取 meta 标签。当浏览器第一次请求页面时，如果不存在标签，则浏览器将会继续全然不知并保持原文件的缓存拷贝。

缓存网络页面的代理服务器，如那些共有的网络服务器提供商（IPS），一般不会自己读取 HTML 文档。网络浏览器可能知道它不应该缓存页面，但在浏览器和网络服务器之间的代理服务器可能不知道——它将继续向客户端传递同样过时的页面。

第二，使用 HTTP 协议来阻止页面缓存从根本上保证了没有网络浏览器或中间代理服务器缓存页面，这样访问者将总能接受到最近的内容。实际上，应该实现的第一个头应是其自身，这是确保页面不被缓存的最好方法。Cache-Control 和 Pragma 头只是为了增加一定程度的保险而添加的。虽然它们不能在所有浏览器或代理上工作，但是 Cache-Control 和 Pragma 头将捕获 Expires 头不像预期那些工作的一些情形，例如，客户端计算机日期设置不正确。

问题 83　如何控制页面背景的打印？

问题阐述

如何控制页面背景的打印呢？





专家解答

在进行 Web 打印时，可以通过以下操作控制是否打印背景颜色和图像。操作步骤如下：

（1）在 IE 窗口中选择“工具”→“Internet 选项”命令。

（2）在弹出的“Internet 选项”对话框中选择“高级”选项卡，在“打印”列表中设置“打印背景颜色和图像”前面的复选框是否勾选，如图 3.8 所示。

（3）如果勾选，则代表打印背景颜色和图像，否则将不打印背景颜色和图像。

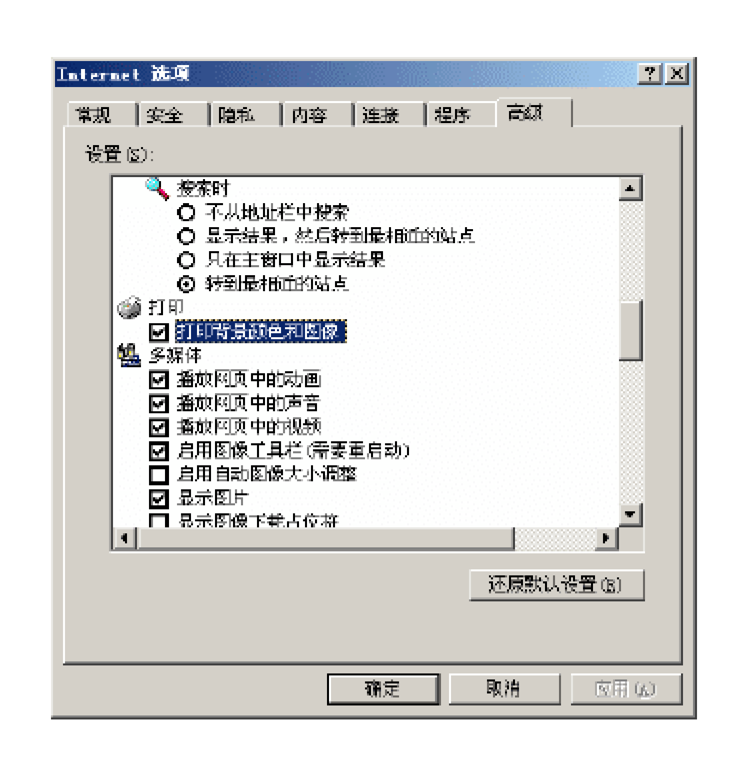


图 3.8　设置打印背景颜色和图像

问题 84　如何实现文字滑动效果？

问题阐述

大家可以看到，在很多网站中的公告栏信息等都采用文字滑动的效果显示，那么，如何实现文字的滑动效果呢？

专家解答

1. 解决思路

文字的滑动效果可以使用<marquee>…</marquee>标签实现。

<marquee>…</marquee>标签是成对出现的，首标签<marquee>和尾标签</marquee>之间的内容就是滚动内容。<marquee>标签的属性包括：behavior、bgcolor、direction、width、height、hspace、vspace、loop、scrollamount、scrolldelay 等，它们都是可选的。

☑　behavior 属性

behavior 属性的参数值为 alternate、scroll、slide 中的一个，分别表示文字来回滚动、单方向循环滚动、只滚动一次。需要注意的是，如果在<marquee>标签中同时出现 direction 和 behavior 属性，那么 scroll 和 slide 的滚动方向将依照 direction 属性中参数的设置。例如：

```
<marquee behavior="alternate">我来回滚动</marquee>
<marquee behavior="scroll">我单方向循环滚动</marquee>
<marquee behavior="scroll" direction="up" height="30">我改单方向向上循环滚动</marquee>
<marquee behavior="slide">我只滚动一次</marquee>
<marquee behavior="slide" direction="up">我改向上只滚动一次了</marquee>
```

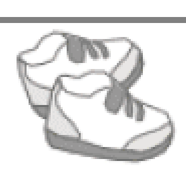
☑　bgcolor 属性

文字滚动范围的背景颜色的参数值是十六进制（形式：#AABBCC 或#AA5566 等）或预定义的颜色名字（如 red、yellow、blue 等）。例如：

```
<marquee behavior=="slide" direction="left" bgcolor="red">我的背景色是红色的</marquee>
```

☑　direction 属性

文字滚动的方向，属性的参数值有 down、left、right、up 共 4 个单一可选值，分别代表滚



动方向向下、向左、向右、向上。例如：

```
<marquee direction="right">我向右滚动</marquee>
<marquee direction="down">我向下滚动</marquee>
```



☑ loop 属性

表示循环的次数，值是正整数，默认为无限循环。

☑ scrollamount 属性

表示运动速度，值是正整数，默认为 6。

☑ scrolldelay 属性

表示停顿时间，值是正整数，默认为 0，单位是毫秒。

☑ valign 属性

表示元素的垂直对齐方式，值可以是 top、middle、bottom，默认为 middle。

☑ align 属性

表示元素的水平对齐方式，值可以是 left、center、right，默认为 left。

☑ height、width 属性

表示运动区域的高度和宽度，值是正整数（单位是像素）或百分数，默认 width=100%，height 为标签内元素的高度。

☑ hspace、vspace 属性

表示元素到区域边界的水平距离和垂直距离，值是正整数，单位是像素。

2. 应用示例

下面使用<marquee>…</MARQUEE>标签制作公告信息滑动显示，结果如图 3.9 所示。

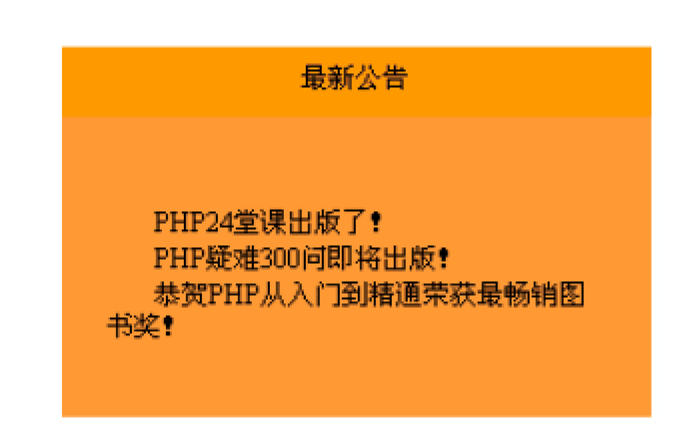


图 3.9　最新公告信息滑动显示效果

关键代码如下：

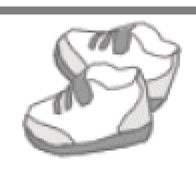
```
<marquee scrollamount="1" scrolldelay="40" direction="up" width="220" height="120" onMouseOver=
"this.stop()" onMouseOut="this.start()">
    <?php
    $board=array('PHP24 堂课出版了！','PHP 疑难 300 问即将出版！','恭贺 PHP 从入门到精通荣获最
畅销图书奖！');
    for($i=0;$i<count($board);$i++){
    ?>
    <div class="divList">         <a target="_blank" class=
"hui12"><?php echo $board[$i];?></a></div>
    <?php
        }
    ?>
    </MARQUEE>
```

问题 85　如何实现自动登录？

问题阐述

所谓自动登录，是指用户第一次成功登录网站后，在一段时间内再次登录这个网站时不再







需要填写用户名和密码，而是可以直接进入。那么这个功能是如何实现的呢？其原理又是怎样的呢？

专家解答

1. 解决思路

本功能实现的关键是 COOKIE 技术。其原理是：当用户第一次登录时，将登录信息存储到 COOKIE 变量中，并且设置 COOKIE 的过期时间；然后当用户再次在本机中登录此网站时，判断 COOKIE 变量是否存在，如果存在则直接登录网站。

其中，创建 COOKIE 应用的是 setcookie()函数，完成 COOKIE 变量的创建、赋值以及过期时间的设置；获取 COOKIE 变量中的值应用的是$_COOKIE[]全局变量。

由于 COOKIE 是 HTTP 头标的组成部分，作为头标必须在页面其他内容之前发送，也必须最先输出，所以在 setcookie()函数之前不能有任何内容输出，即使是一个 HTML 标记、一个 echo 语句甚至一个空行都会导致程序出错，这就是 setcookie()函数，下面是它的语法：

```
bool setcookie(string name[,string value[,int expire[, string path[,string domain[,int secure]]]]])
```

setcookie()函数的参数说明如表 3.1 所示。

表 3.1　setcookie()函数的参数说明

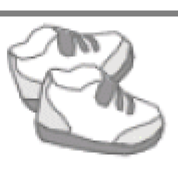
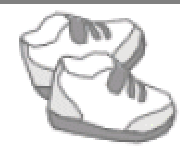
参　数	说　明	举　例
name	COOKIE的变量名	可以通过$_COOKIE['cookiename ']调用变量名为cookiename的COOKIE
value	COOKIE变量的值，该值保存在客户端，不能用来保存敏感数据	可以通过$_COOKIE['values ']获取名为values的值
expire	COOKIE的过期时间，expire是标准的UNIX时间标记，可以用time()函数或mktime()函数获取，单位为秒	如果不设置COOKIE的过期时间，那么COOKIE将永远有效，除非手动将其删除
path	COOKIE在服务器端的有效路径	如果该参数设置为“/”，则它在整个domain内有效；如果设置为“/10.9”，则它在domain下的/10.9目录及子目录内有效。默认是当前目录
domain	COOKIE有效的域名	如果要使COOKIE在mrbccd.cn域名下的所有子域都有效，则应设置为mrbccd.cn
secure	指明COOKIE是否仅通过安全的HTTPS，值为0或1	如果值为1，则COOKIE只能在HTTPS连接上有效；如果值为默认值0，则COOKIE在HTTP和HTTPS连接上均有效

2. 应用示例

本示例实现一个自动登录的功能。其具体步骤如下：

（1）创建 index.php 文件，创建表单，添加用户名、密码文本框和保存时间选择单选按钮。最关键的是编写 PHP 脚本，通过$_COOKIE 获取 COOKIE 中存储的用户名和密码数据，并将其作为用户名和密码的默认值。其关键代码如下：

```
<form id="form1" name="form1" method="post" action="index_ok.php">
    <input id="lgname" name="name" value="<?php echo $_COOKIE['name'];?>" type="text" class="txt"/>
    <input id="lgpwd" name="pwd" value="<?php echo $_COOKIE['pwd'];?>" type="password" class="txt"/>
```

```
        <input name="times" type="radio" value="3600" checked="checked" />1 小时
        <input type="radio" name="times" value="86400" />1 天
        <input type="image" name="imageField" src="images/dl.gif" />
        <input type="image" name="imageField2" onclick="form.reset();return false;" src="images/cz.gif" />
    </form>
```



（2）创建 index_ok.php 文件，通过$_POST 方法获取表单中提交的数据，验证用户输入的用户名和密码是否正确。如果正确，则通过 setcookie()函数创建 COOKIE，存储用户名和密码，并根据表单提交的时间设置 COOKIE 过期时间，同时跳转到 main.php 页面；如果不正确，则给出提示信息，并跳转到 index.php 页面。其关键代码如下：

```
<?php
if (! empty ( $_POST ['name'] ) and ! empty ( $_POST ['pwd'] )) {      //判断用户名和密码是否为空
    if ($_POST ['name'] == "mr" && $_POST ['pwd'] == "mrsoft") {
        setcookie ( "name", $_POST ['name'], time () + $_POST ['times'] );
                                                    //设置 COOKIE 有效时间为 1 小时
        setcookie ( "pwd", $_POST ['pwd'], time () + $_POST ['times'] );
                                                    //设置 COOKIE 有效时间为 1 小时
        echo "<script>alert('succeed!');window.location.href='main.php';</script>";
    } else {
        echo "<script>alert('false!');window.location.href='index.php';</script>";
    }
} else {
        echo "<script>alert('NULL');window.location.href='index.php';</script>";
}
?>
```

（3）创建 main.php 文件，首先根据$_COOKIE 获取的 COOKIE 值判断用户是否具有访问权限，如果有则可以看到本页内容，否则将给出提示信息，跳转到 index.php 页面。其关键代码如下：

```
<?php
if($_COOKIE[name]==""){          //根据 COOKIE 的值，判断浏览者是否具有访问该页面的权限
    echo "<script>alert('您不具有访问该页面的权限！ '); window.location.href='index.php';</script>";
}else{                           //如果正确则输出主页内容
?>
<!—省略了 HTML 内容-->
<?php
}
?>
```

运行本示例，如果是第一次登录，则需要填写用户名和密码才能够登录；如果不是第一次登录，则不需要输入用户名和密码，因为$_COOKIE 会从 COOKIE 中读取到用户名和密码，用户直接单击登录按钮即可。登录成功后进入 main.php 页面，运行结果如图 3.10 所示。

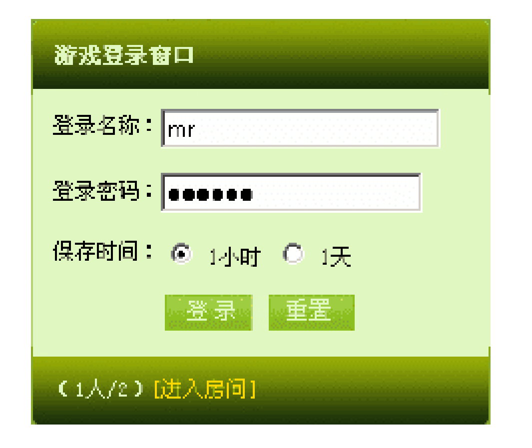
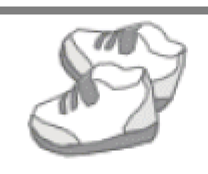




图 3.10　进入网站主页



专家点评

在使用 COOKIE 时，如果设置 COOKIE 的过期时间，那么浏览器将会保存 COOKIE 数据，即使用户重新启动计算机，只要未过期，COOKIE 数据就一直有效。

问题 86　如何统计用户在线时间？

问题阐述

所谓用户在线时间就是当用户登录到页面开始，到关闭浏览器结束的时间差，而 COOKIE 的生命周期在默认情况下关闭浏览器时自动删除。这非常符合在线统计时间的特点，是很多服务类网站优先选择的统计手段。

专家解答

1. 解决思路

统计用户在线时间的原理：通过 COOKIE 保存用户登录到页面的时间戳和离开网站时的时间戳，将这两个时间戳做减法运算，所得到的就是用户的在线时间。其具体做法如下：

用户登录时间：

```
$str = time();
setcookie('times',$str);
```

用户退出时间：

```
$string = time()-$_COOKIE['times'];
echo "在线时间：".date("H:i:s",$string)."秒！"."<br>";
```

例如，通过创建 COOKIE，保存用户登录到页面的时间戳，在用户单击退出按钮时计算用户在线时间。运行结果如图 3.11 所示。

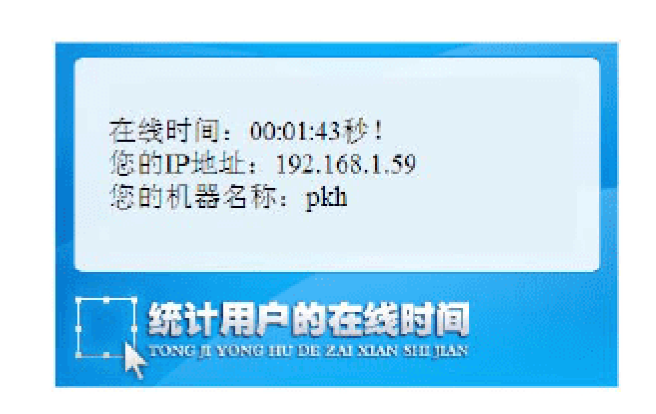
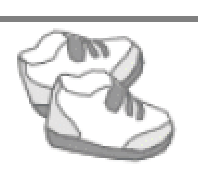


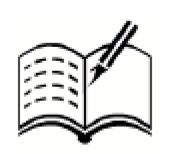
图 3.11　统计用户的在线时间

实现步骤如下：

（1）在 index.php 文件中创建 COOKIE，存储用户的登录时间戳。其关键代码如下：

```
<?php
    include_once 'conn/conn.php';                    //执行连接数据库的操作
    $str = time();
    setcookie('times',$str);
    if($_COOKIE[name]!="" and $_COOKIE[id]!=""){
        $sql = "select * from tb_login where name = '".$_COOKIE[name]."' and login_id = '".$_COOKIE[id]."'";
        $num = $conne->getRowsNum($sql);          //返回查询结果
        $conne->close_rst();                      //关闭数据库
        if($num != 0 or $num != ''){
            header("Location: default.php");
        }else{
?>
```





（2）创建 information.php 文件，应用系统的当前时间戳减去 COOKIE 中存储的登录时间戳，即获取到系统的在线时间的时间戳，并通过 date()函数对这个时间戳进行格式化，并输出用户的在线时间，同时输出登录用户的 IP 和机器名称。其关键代码如下：

```
<?php
		$string = time()-$_COOKIE['times'];
		echo "在线时间：".date("H:i:s",$string)."秒！"."<br>";
		echo "您的 IP 地址：".getenv('REMOTE_ADDR')."<br>";
		echo "您的机器名称：".gethostbyaddr(getenv('REMOTE_ADDR'));
?>
```

问题 87　如何限制用户访问网站的时间？

问题阐述

默认情况下，COOKIE 生命周期是以关闭浏览器为基准的，也就是说，如果用户不关闭浏览器并且没有对 COOKIE 进行设置，则 COOKIE 永远也不会过期失效。假如用户在互联网发布网站，有成百上千的用户浏览这个网站，并且在线浏览用户数量一直增加。如果不对用户访问网站的时间进行限制，则结果只能是服务器资源耗尽，网站瘫痪。那么如何限制用户访问网站的时间呢？

专家解答

1. 解决思路

首先通过 setcookie()函数创建 COOKIE 并设置 COOKIE 的有效时间，然后通过判断 COOKIE 变量是否存在和 COOKIE 变量的值是否为空来控制访问网站的时间。其关键代码如下：

```
<?php
session_start();                                //初始化 SESSION 变量
$session_id=session_id();                       //获取 SESSION ID
setcookie("start",$session_id,time()+10);  //定义 COOKIE 变量，设置 COOKIE 的有效时间是 10 秒
?>
```

在上述的方法中，定义 COOKIE 的过期时间为 10，即网站的访问时间为 10 秒，其原理就是如此，至于在实际的应用中，设置网站的访问时间为多长，可由用户自行决定。

2. 应用示例

本示例通过设置 COOKIE 限制用户访问网站的时间，运行结果如图 3.12 和图 3.13 所示。



图 3.12　COOKIE 未失效，访问网站



图 3.13　COOKIE 失效





实现步骤如下：

（1）创建 index.php 文件。首先，初始化 SESSION 变量，获取 SESSION_ID，然后，通过 setcookie()函数创建 COOKIE，并将 SESSION_ID 作为 COOKIE 值，同时设置 COOKIE 的有效时间为 10 秒。

（2）在页面中通过判断 COOKIE 变量的值是否为空来限制用户访问网站的时间。其关键代码如下：

```
<?php
    if(isset($_COOKIE['start'])|| $_COOKIE['start']==$session_id){
?>
……//省略了网页中的代码
<?php
    }else{
    echo "<h1 style ='color:red;'>您访问网站的时间到了</h1>";
}
?>
```

问题 88　如何掌控登录用户的权限？

问题阐述

在论坛中，如果是管理员，则可以发布公告信息；如果是普通用户，则不可以发布公告信息。那么程序中是如何对登录用户的权限进行判断的呢？

专家解答

1. 解决思路

在用户登录成功后，将登录用户信息存储到一个 SESSION 变量中，这个 SESSION 变量值在网站中（本次会话结束之前）一直有效，那么就可以根据这个 SESSION 变量的值判断用户的权限。

创建一个 SESSION 变量，首先要通过 session_start()函数初始化 SESSION 变量，然后通过预定义变量$_SESSION[]直接为变量赋值。其实现代码如下：

```
session_start();
$_SESSION["name"] = "value";
```

通过预定义变量$_SESSION[]定义 SESSION 变量。“name”为设置变量的名称，“value”表示设置变量的值。

2. 应用示例

通过 SESSION 变量掌控登录用户的权限。如果是管理员登录，则可以添加公告信息；而如果是普通用户，则只可以浏览论坛中的帖子。其运行结果如图 3.14 和图 3.15 所示。

上述是以管理员身份进行登录，进入 main.php 主页后，可以单击“公告”超链接，发布公告信息；如果是以普通用户身份登录，那么在 main.php 主页中，“公告”超链接是不可以点击的，因为普通用户不具备发布留言的权限。





图 3.14 管理员登录



图 3.15 管理员页面

实现步骤如下：

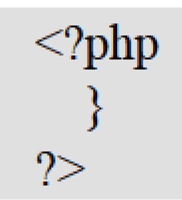
（1）创建 index.php 文件，添加表单，设计用户登录页面，并获取表单提交的用户名和密码，在本页中对用户提交的数据进行判断。首先，初始化 SESSION 变量，判断用户提交的值是否为空，同时判断如果登录用户是管理员，则为 SESSION 变量 type 赋值为 0；如果登录用户是普通用户，则为 SESSION 变量 type 赋值为 1。其关键代码如下：

```
<?php
    session_start();
    if($_POST['sub']==true){
        if($_POST['user']!="" && $_POST['pwd']!=""){        //判断用户名和密码是否为空
            if($_POST['user']=="mr" && $_POST['pwd']=="mrsoft"){   //判断是否是管理员
                $_SESSION['type'] = 0;                        //为 SESSION 变量赋值为 0
                echo "<script>alert('管理员登录成功！'); window.location.href='main.php';</script>";
            }else{
                $_SESSION['type'] = 1;      //如果不是管理员，则为 SESSION 变量赋值为 1
                echo "<script>alert('普通用户登录成功！'); window.location.href='main.php';</script>";
            }
            $_SESSION['user'] = $_POST['user'];   //将登录用户名存储到 SESSION 变量 user 中
            $_SESSION['pwd'] = $_POST['pwd'];    //将登录密码存储到 SESSION 变量 pwd 中
        }else{
            echo "<script>alert('用户名和密码不能为空！'); window.location.href='index.php';</script>";
        }
    }
?>
```

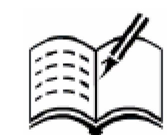
（2）创建 main.php 文件，编写网站的主页面。首先，初始化 SESSION 变量，然后，根据 SESSION 变量 type 的值定义页面的输出内容，如果 SESSION 变量的值为 0，则说明登录用户是管理员，即具备发布公告的权限；如果 SESSION 变量的值为 1，则说明登录用户是普通用户，即不具备发布公告的权限，也就是说控制发布公告的超链接不输出。其关键代码如下：

```
<?php
session_start();                              //初始化 SESSION 变量
?>
<?php
  if($_SESSION['type']=="0"){//根据 SESSION 变量 type 的值决定是否输出发布公告的超链接
?>
<area shape="rect" coords="330,64,385,100" href="mes.php" />
```





```
<?php
  }
?>
```



Note

（3）创建 mes.php 文件，编写发布公告信息的页面，根据 SESSION 变量 type 的值决定用户是否具有访问此页面的权限。

专家点评

SESSION 生命周期从用户访问页面开始到断开与网站的链接为止。用户在每一次链接时，PHP 都会自动生成一个唯一的 SESSION ID 以标识当前用户与其他用户。

问题 89　如何让 SESSION 永久有效？

问题阐述

在网站中常常会出现这样的情况，在以一个管理员的身份登录网站后，在认真地回复客户的问题时，时间过去大概半个小时之后，系统提示管理员需要重新登录。出现此问题的原因是 SESSION 变量中存储的用户已经过期，那么该如何解决这个问题呢？

专家解答

1. 解决思路

要让 PHP 中的 SESSION 永不过期，解决方案可以考虑两个方面，第一是对配置文件（php.ini）进行修改，此方法只需修改配置文件，而不必修改程序，非常理想，但若用户不具备修改配置文件的权限，那么就只能考虑第二种方案了，即通过对程序的修改，实现 SESSION 永不过期。

说明：

这里所指的 SESSION 永不过期，不包括手动销毁 SESSION。

2. 通过修改配置文件实现 SESSION 永不过期

打开 php.ini 配置文件，定位到如下 3 个位置，并对它们的值进行修改：

```
session.use_cookies = 1
```

设置它的值为 1，通过 COOKIE 来传递 SESSION ID。

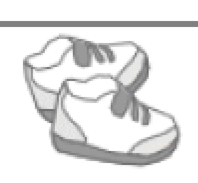
```
session.cookie_lifetime = 0
```

它的作用是设置 SESSION ID 在客户端 COOKIE 储存的时间，默认值是 0，表示浏览器关闭后 SESSION ID 就销毁。将它的值设置的大一些也就延长了 SESSION 的有效时间。

```
session.gc_maxlifetime = 1440
```

它是 SESSION 数据在服务器端储存的时间，如果超过这个时间，那么 SESSION 数据就自动删除。同样将它的值也设置的大一些也就延长了 SESSION 在服务器端的有效时间。

在 php.ini 文件中，完成对上述 3 个值的修改后，SESSION 就可以实现永久存储了，前提是只要用户设置的值足够大。





Note

3. 在程序中实现 SESSION 永不过期

如果用户不具备修改 php.ini 配置文件的权限，那么就只能通过修改程序实现 SESSION 的永久存储了。通过 COOKIE 存储 SESSIONID，其具体方法如下：

```
<?php
session_start();                                              //启动 SESSION
$_SESSION['cou'];                                             //注册 SESSION 变量 Count
isset($PHPSESSID)?session_id($PHPSESSID):$PHPSESSID = session_id();
        //如果设置了$PHPSESSID，就将 SessionID 赋值为$PHPSESSID，否则生成 SESSION ID
$_SESSION['cou']++;                                           //变量 cou 加 1
setcookie('PHPSESSID', $PHPSESSID, time()+36000000);          //储存 SESSION ID 到 Cookie 中
echo $cou;                                                    //显示 SESSION 变量 cou 的值
?>
```

问题 90　SESSION 购物车

问题阐述

SESSION 购物车，顾名思义，其主要就是应用 SESSION 变量来实现的。那么，如何通过 SESSION 变量来实现购物车功能呢？

专家解答

1. 解决思路

购物车的操作流程：首先，登录到网站中浏览商品。然后，购买指定的商品，进入购物车页面中，可以实现很多的操作，包括：更改商品数量、删除商品、清空购物车、继续购物等。最后，填写收货人信息，生成订单。其操作流程如图 3.16 所示。

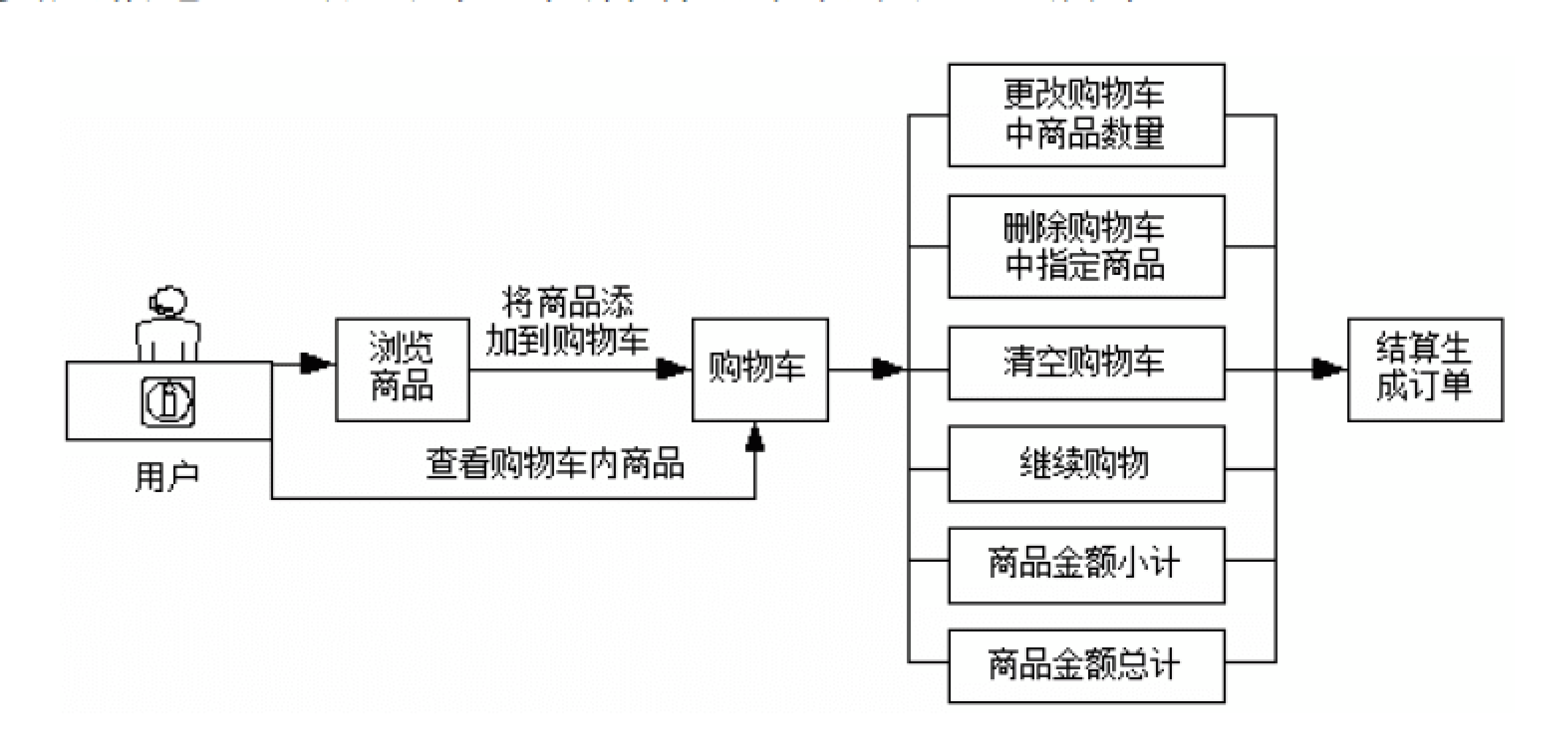
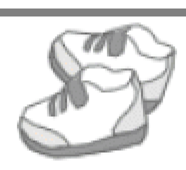


图 3.16　购物车操作流程

这里所围绕的购物车，其实就是在用户登录网站时为其创建的两个 SESSION 变量，其中一个存储商品 id，并且以@符号分隔不同 id；另一个对应存储商品的购买数量，同样以@符号进行分隔。

而针对购物车进行的各种操作，也都是围绕这两个 SESSION 变量中的值进行的。添加商品，就是向 SESSION 变量中添加新的 id 和对应的数量值；更新商品购买数量，则是根据商品的 id，更新对应商品的数量；删除商品，同样是根据商品的 ID，删除对应商品数量；清空购



物车，则是直接为这两个 SESSION 变量赋空值。这就是购物车的基本原理，至于将其付诸实践，相对会复杂一些，会涉及字符串与数组之间的转换，以及数组中指定元素值的更新和删除等。

2. 应用示例

购物车是在网上购物时使用的一个临时存储商品的“车辆”。在本示例中，我们将介绍 SESSION 购物车的实现方法，其运行结果如图 3.17 所示。

应用 SESSION 技术开发购物车的具体步骤如下：

（1）创建 index.php 页面，实现商品展示功能。设计原理是：从数据库中读取商品信息，将商品信息在页面中进行分栏、分页显示，并且为商品设置购买和查看购物车的超链接，运行结果如图 3.18 所示。





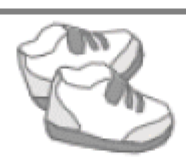
图 3.17　SESSION 购物车



图 3.18　购物车商品展示页面

（2）创建 by_commodity.php 文件，实现购买功能，即向购物车中添加商品。首先创建一个购物车，然后判断购物车中是否为空，如果为空，则将商品展示页中“购买”超链接传递的商品 ID（$_GET[id]）和数量添加到购物车中；如果不为空，则判断添加商品的 ID 是否在购物车中已经存在，如果存在则不能重复添加，否则将商品 ID 添加到购物车中，最后跳转到 shopping.php 购物车页面，其代码如下：

```
<?php
session_start();                                          //初始化 SESSION 变量
header ( "Content-type: text/html; charset=UTF-8" );      //设置文件编码格式
if($_SESSION["goodsid"]=="" && $_SESSION["goodsnum"]==""){//判断 SESSION 变量是否为空
    $_SESSION["goodsid"]=$_GET["id"]."@";                 //如果 SESSION 变量为空，则为其赋值
                                                            为商品的 ID，并以@分隔
    $_SESSION["goodsnum"]="1@";    //如果 SESSION 变量为空，则为其赋值为 1，并以@分隔
}else{                             //如果 SESSION 变量不为空
     $array=explode("@",$_SESSION["goodsid"]);           //则以@为分隔符，将 SESSION 变量中
                                                            的数据写入数组中
     if(in_array($_GET["id"],$array)){                    //判断数组中是否存在指定的 ID
       echo "<script>alert('该商品已经被放入购物车！ ');history.back();</script>";
       exit;
     }
    //如果数组中不存在指定的 ID，则说明该商品还未被放入购物车中
    $_SESSION["goodsid"].=$_GET["id"]."@";                //将该商品添加到购物车中
    $_SESSION["goodsnum"].="1@";                          //更改商品数量
}
echo "<script>window.location.href='shopping_car.php';</script>";
?>
```



（3）创建 shopping.php 购物车页面，对购买商品进行统计，包括购买数量、每种商品的金额和总金额，并且实现删除购物车中商品、更改购物车中商品数量和清空购物车的功能，同时设置“继续购买”的超链接，其代码可参考本书配套资源。


Note

（4）创建 change_commodity_counts.php 文件，实现更改购物车中商品的数量。首先获取 shopping.php 页面表单中提交的商品数量和商品 ID，然后通过正则表达式验证商品数量是否为正整数，最后根据商品 ID 找到指定的商品，更改商品数量，并返回购物车页面，其代码如下：



```
<?php
session_start();
header ( "Content-type: text/html; charset=UTF-8" );          //设置文件编码格式
$id=$_POST["id"];                                              //获取商品 ID
$num=$_POST["goodsnum"];                                       //获取商品数量
$preg="/^[0-9]*[0-9]$|^[0-9]*[0-9]$/";                          //编写正则表达式
if($num==""){                                                  //判断提交的值是否为空
      echo "<script>alert('数量不能为空!');history.back();</script>";
      exit;
}else if(!preg_match($preg,$num,$str)){                        //判断提交的数据是否是正整数
        echo "<script>alert('数量只能为正整数!');history.back();</script>";
        exit;
}
$arrayid=explode("@",$_SESSION["goodsid"]);
$arraynum=explode("@",$_SESSION["goodsnum"]);
$key=array_search($id,$arrayid);                               //在数组中搜索给定的值，如果成
                                                                 功则返回相应的键名
$arraynum[$key]=$num;                                          //更改商品数量
$_SESSION["goodsnum"]=implode("@",$arraynum);                  //更改商品数量
echo "<script>window.location.href='shopping_car.php';</script>";
?>
```

（5）创建 delete_commodity.php 文件，删除购物车中指定的商品。首先获取超链接中传递的商品 ID，然后将 SESSION 中的字符串数据转换成数组，并根据商品 ID 获取指定要删除的商品的键名，最后根据键名为指定的数组赋空值，并重新将数组添加到购物车中。

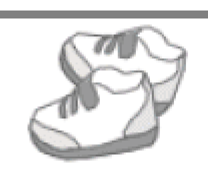
（6）创建 clear_shopping_car.php 文件，清空购物车，即将 SESSION 变量$_SESSION ["goodsid"]和$_SESSION["goodsnum"]赋值为空。其代码如下：

```
<?php
session_start();                                                //初始化 SESSION 变量
header ( "Content-type: text/html; charset=UTF-8" );           //设置文件编码格式
$_SESSION["goodsid"]="";                                        //为购物车的 ID 赋值为空
$_SESSION["goodsnum"]="";                                       //为购物车数量赋值为空
echo "<script>window.location.href='shopping_car.php';</script>"; //完成清空操作，跳转到购物页面
?>
```

专家点评

在应用 SESSION 变量实现购物车功能时，最容易出现的一个错误就是忘记初始化 SESSION 变量，如果在文件中没有初始化 SESSION 变量，那么在页面之间就获取不到 SESSION 变量中存储的数据。

清空购物车中的商品可以采用直接为 SESSION 变量赋空值的方法，也可以直接应用 session_destroy()函数删除 SESSION，或者应用 unset()函数删除指定的 SESSION 变量。



问题 91　header()函数都有哪些功能？



问题阐述

通过 header()函数可以发送一个原始的 HTTP 头，那么通过它具体能实现什么样的功能呢？

专家解答

1. HTTP 头

服务器发送回来的 HTTP 响应包含以下信息：用于识别响应主体内容的头（header），发送响应的服务器，响应消息有多少字节，以及响应何时发出等。PHP 和 Apache 已经完成对头信息的处理：将文档识别为 HTML 和计算 HTML 页面的长度等。绝大多数 Web 程序不需要自己设置头。但是，如果想让服务器返回的不是 HTML，或者想设置页面的过期时间，重定向客户端浏览器到另一个地址，或是产生一个特定的 HTTP 错误，就需要使用 header()函数。

2. header()函数

header()函数的语法如下：

```
void header ( string string [, bool replace [, int http_response_code]] )
```

header()函数的参数说明如表 3.2 所示。

表 3.2　header()函数的参数说明

参　　数	说　　明
string	必要参数。输入的头部信息
replace	可选参数。指明是替换掉前一条类似的标头还是增加一条相同类型的标头。默认为替换，但如果将其设为false，则可以强制发送多个同类标头
http_response_code	可选参数。强制将HTTP响应代码设为指定值，此参数是php5.3.0以后添加的

设置 header 一定要在生成主体内容之前完成，这意味着所有 header()（或 setcookie()，如果你想设置 COOKIE）要在文件的最前面，在<html>标签之前。

3. 设置文档类型

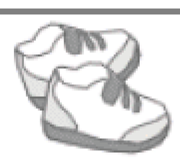
Content-Type 头指定被返回文档的类型。它通常是“text/html”，指明它是一个 HTML 文档，但还有其他一些有用的文档类型，例如，“text/plain”让浏览器强制性地将内容当作纯文本来处理。这个类型就类似于自动的“查看源代码”，它在调试时很有用。

例如，应用 header()函数，设置文档类型为 image/png，其代码如下：

```
header("Content-type:image/png");      //发送头部信息，生成 png 的图片文件
```

4. 重定向

通过 header()函数可以向浏览器发送一个新的 URL，并让浏览器转向到该地址。这样的重定向（redirection）操作只需要通过设置 Location 头即可。例如，通过 header()函数重定向到 http://www.mrbccd.com，其代码如下：



```
header("Location: http://www.mrbccd.com");
```

重定向操作更倾向于绝对路径，如果提供相对的 URL（如“/index.php”），则重定向会在服务器内部进行。这种方法很少用，因为浏览器并不知道它得到的页面是否是所请求的。如果在新的文档中存在相对 URL，则浏览器会将它们解释成相对于所请求的文档，而不是被发送的文档。

Note

5. 设置过期时间

服务器可以显式地通知浏览器（或者那些存在于服务器的浏览器之间的代理服务器缓存）文档的过期时间。代理服务器和浏览器缓存在过期之前可保持文件，或提前结束它。重新载入一个被缓存的页面不需要和服务器进行通信。但是尝试获取一个已经过期的文档就需要与服务器取得联系。

为一个文档设置过期时间，应用的是 Expires 头：

```
header("Expires:    Mon,    08    Jul    2011    08:08:08    GMT");
```

例如，控制文档在页面生成后的两小时后过期。使用 time()和 gmstrftime()函数生成过期日期字符串：

```
<?php
$now=time();                                                    //获取系统当前时间戳
$then=gmstrftime("%a,%d %b %Y %H:%M:%s GMT",$now + 60*60*2);   //格式化时间
header("Expires:$then");                                        //定义文档过期时间
?>
```

6. HTTP 认证

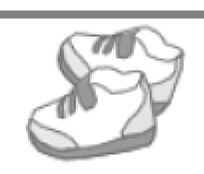
HTTP 认证（HTTP authentication）通过请求的 header 和响应状态来工作。浏览器可以将用户名和密码放在请求的头中发送。如果认证凭证（credential，即指用户名和密码）未发送或不匹配，服务器将发送一个“401 Unauthorized”响应并通过 WWW 认证头来确定当前认证的区域（realm）（一个字符串，诸如“Mary's Pictures”或“Your Shopping Cart”）。这通常会导致浏览器弹出一个“Enter username and password for…”对话框，且该页面会重新请求更新头中的认证凭证。

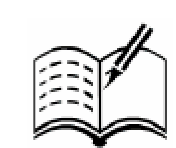
为了用 PHP 来处理认证，可检查用户名和密码（$_SERVER 数组中的 PHP_AUTE_USER 和 PHP_AUTH_PW 两个元素）并调用 header()函数来设置区域，然后发送一个“401 Unauthorized”响应。其关键代码如下：

```
header('WWW-Authenticate:Basic realm="Top Secret Files"');
header("HTTP/1.0 401 Unauthorized");
```

另外，header()函数还可以强制客户端每次访问页面时获取最新资料，而不是使用存在于客户端的缓存。其关键代码如下：

```
header("Expires: Mon, 08 Jul 2011 08:08:08 GMT");            //设置页面的过期时间（用格林威
                                                               治时间表示）
header("Last-Modified:".gmdate("D, d M Y H:i:s")."GMT");      //设置最后更新日期（用格林威治
                                                               时间表示），使浏览器获取最新资料
header("Cache-Control: no-store,no-cache, must-revalidate");  //控制页面不使用缓存
header("Cache-Control: post-check=0,pre-check=0",false);      //控制页面不使用缓存
header("Pragma: no-cache");          //参数（与以前的服务器兼容），即兼容 HTTP1.0 协议
```





7. 文件下载

根据服务器返回的头信息，应用 header 函数可以完成文件的下载操作。其关键代码如下：

```
header("Content-type:application/octet-stream");              //输出 MIME 类型
header("Accept-ranges:bytes");                                 //接受的范围单位
header("Accept-length:".filesize($path));                      //文件长度
header("Content-Disposition:attachment;filename=".$filename);  //默认时文件保存对话框中的文件
                                                               名称
```

例如，通过 header()函数生成 Excel 格式的下载文件，其关键代码如下：

```
<?php
header("Content-type:application/vnd.ms=excel");              //设置 HTTP 标头
header("Content-Disposition:filename=会议报表.xls");           //定义下载文件名称
?>
```

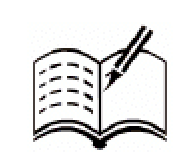


数据库操作

- 问题92　MySQL数据库开发常识有哪些?
- 问题93　MySQL的常用命令有哪些?
- 问题94　MySQL数据存储引擎InnoDB和MyISAM的优势及区别有哪些?
- 问题95　如何在MySQL中实现多关键字查询?
- 问题96　如何在MySQL中修改用户密码和创建权限用户?
- 问题97　在MySQL中两种索引Hash与B-Tree的区别是什么?
- 问题98　应用PHP操作MySQL数据库时应该注意哪些问题?
- 问题99　应用PHP向MySQL插入数据时，如何避免产生乱码?
- 问题100　MySQL常见错误代码有哪些?
- 问题101　在MySQL中，如何实现多表查询?
- 问题102　在MySQL中，如何实现内连接查询?
- 问题103　在MySQL中，如何实现外连接查询?
- 问题104　在MySQL中，如何实现嵌套查询?
- 问题105　如何连接远程MySQL服务器?
- 问题106　如何实现MySQL数据库的双机热备份?
- 问题107　解决MySQL语句错误的最佳方案是什么?
- 问题108　如何执行数据的批量添加?
- 问题109　如何规范地定义数据表中的字段名称?
- 问题110　如何查询指定时间段的数据?
- 问题111　如何按月查询统计数据?
- 问题112　如何实现在删除数据前给予提示信息?
- 问题113　如何查询日期型数据?
- 问题114　如何使用左连接（left outer join）查询数据?
- 问题115　如何使用右连接（right outer join）查询数据?
- 问题116　如何对统计结果进行排序?
- 问题117　如何使用聚集函数SUM对学生成绩进行汇总?
- 问题118　如何使用聚集函数MIN求利润最少的商品?
- 问题119　如何使用聚集函数MAX求销售利润最高的商品?
- 问题120　如何使用INSERT(s1,x,len,s2)函数替换指定字符?
- 问题121　如何使用UPPER(s),UCASE(s)函数将字母变成大写?
- 问题122　如何利用CSS样式实现分页打印?
- 问题123　如何使用RTRIM(s)函数去掉字符串结尾处的空格?
- 问题124　MySQL数据库的备份和恢复
- 问题125　如何在MySQL中创建存储过程?
- 问题126　如何将Access数据导入SQL Server数据库中?
- 问题127　如何解决PHP不支持SQL Server的问题?
- 问题128　如何连接带密码的Access数据库?
- 问题129　MySQL和Server 2000数据库中数据类型的转换
- 问题130　PHP如何连接SQL Server数据库?



问题 92　MySQL 数据库开发常识有哪些？



问题阐述

MySQL 数据库是目前程序员应用较多的数据库之一，选择其作为数据库应掌握哪些基本常识？

专家解答

MySQL 数据库是目前比较流行的数据库之一，作为程序员应该掌握与其相关的一些常识，如存储引擎的选择、索引的设计和使用以及大批量插入时 SQL 语句的优化等，可以帮助大家更好地使用 MySQL 数据库。这里主要强调数据存储引擎的选择以及索引的设计和使用。

1. 存储引擎的选择

（1）MyISAM 不支持事务，且不支持外键，优点是访问速度高，批量插入速度快。假设大量的操作由 select、insert 组成，建议采用该存储引擎。但是在实际应用中，可能出现由于批量插入过于频繁而导致数据量到达一定级别，从而出现表损坏的情况。

（2）InnoDB 支持事务处理，但是相对于前者其处理效率稍低，并且其索引及数据也更占用磁盘空间。在存储一些关键数据并需要对其进行事务操作时，可以选择 InnoDB。其前提是没有太大的访问量。

2. 索引的设计和使用

没有索引的表会在查询过程中为用户带来不必要的麻烦，除非数据表中数据量小。下面简要分析索引的设计和使用。

（1）索引通常是设置 where 子句中的列，如果用户设置 select 后的列，这是没有任何意义的。当然，如果用户需要对某列进行排序，order by 后的列也是可以建成索引的。

（2）使用唯一索引，主键就是最好的例子，假设索引列大都是重复的，如性别，这样的索引并不会加快搜索速度。

（3）只要有可能，就要尽量限定索引的长度，例如，索引列为 char(100)，在其前 10 个字符大部分都是唯一的，应设置索引的长度为 10，使用短索引可以加快查询速度，并节省硬盘空间。

（4）索引的左前缀特性，联合索引实质上也是建立了多个索引，那么是建立联合索引好还是分别建多个索引好呢？显然前者更好，利用左前缀特性，只要联合索引的最左的列被用到，那么索引都将被使用。

（5）最后需要声明的是，不要过度使用索引，索引越多，插入的速度就越慢，尤其是在数据量庞大时，同时，大量的索引也将耗费很大的硬盘空间，造成不必要的浪费。

专家点评

存储引擎的选择与索引的设计及使用只是数据库开发常识中的一部分。为了使数据库系统能够发挥其最大功效，用户应选择最适合的存储引擎，并通过索引的设计来优化数据表的查询和提高整个数据库系统的性能。



问题93　MySQL的常用命令有哪些？



Note

问题阐述

在MySQL中，有哪些常用的命令是用户必须了解和掌握的？

专家解答

对于常用的MySQL命令，无非是“增、删、改、查”操作，但是，在不依靠MySQL第三方可视管理软件的情况下，只掌握MySQL的查询和查看命令是远远不够的。所以，用户还必须掌握其他常用的命令。

1. 使用SHOW语句找在服务器上当前存在的数据库

```
SHOW DATABASES;
```

2. 创建一个数据库MYSQLDATA

```
CREATE DATABASE MYSQLDATA;
```

3. 选择你所创建的数据库

```
mysql> USE MYSQLDATA;
```

4. 查看当前数据库表

```
  SHOW TABLES;
```

5. 创建数据表

```
CREATE TABLE MYTABLE (name VARCHAR(20), sex CHAR(1));
```

6. 查看数据表结构

```
DESCRIBE MYTABLE;
```

7. 向数据表中添加记录

```
insert into mingri values (”mr”,”mrsoft”);
```

8. 以文本形式将数据装入数据表中

```
LOAD DATA LOCAL INFILE “D:/mysql.txt” INTO TABLE MYTABLE;
```

9. 导入.sql文件命令

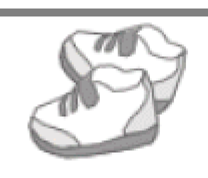
```
use database;
source d:/mysql.sql;
```

10. 删除数据表

```
drop TABLE MYTABLE;
```

11. 清空数据表

```
delete from MYTABLE;
```





12. 更新数据表

```
update MYTABLE set sid="3" where name='mrsoft';
```

专家点评

熟练掌握常用的 MySQL 命令就如同一名优秀的士兵必须熟练掌握手中武器的使用方法一样。在 MySQL 日常操作中，熟练掌握这些常用的操作命令是十分必要的，可以更好地帮助用户使用和学习 MySQL 数据库。同样，作为一名数据库管理人员或程序开发人员，熟练操作 MySQL 数据库更是一项基本的技能。

问题 94　MySQL 数据存储引擎 InnoDB 和 MyISAM 的优势及区别有哪些？

问题阐述

常用的 MySQL 数据存储引擎有 MyISAM（默认）和 InnoDB 等类型，这两种类型的存储引擎有什么区别？其各自优势又是什么？

专家解答

1. MyISAM 存储引擎

MyISAM 存储引擎是 MySQL 中常见的存储引擎，曾是 MySQL 的默认存储引擎。MyISAM 存储引擎是基于 ISAM 存储引擎发展起来的，它解决了 ISAM 的很多不足，同时它还增加了很多有用的扩展。

MyISAM 存储引擎的表存储成 3 个文件。文件的名字与表名相同，其扩展名包括 frm、MYD 和 MYI。

- ☑ frm：存储表的结构。
- ☑ MYD：存储数据，是 MYData 的缩写。
- ☑ MYI：存储索引，是 MYIndex 的缩写。

基于 MyISAM 存储引擎的表支持 3 种不同的存储格式，包括静态型、动态型和压缩型。

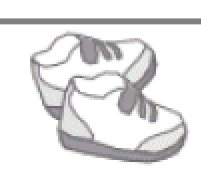
（1）MyISAM 静态型

如果所有表列的大小都是静态的（即不使用 xBLOB、xTEXT 或 VARCHAR 数据类型），则 MySQL 就会自动使用静态 MyISAM 格式。使用这种类型的表性能非常高，因为在维护和访问以预定义格式存储的数据时需要很低的开销，但是，这项优点要以空间为代价，因为每列都需要分配给该列的最大空间，而无论该空间是否真正地使用。

（2）MyISAM 动态型

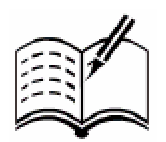
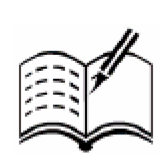
如果有表列（即使只有一列）定义为动态的（使用 xBLOB、xTEXT 或 VARCHAR），则 MySQL 就会自动使用动态格式。虽然 MyISAM 动态表占用的空间比静态格式所占空间少，但空间的节省带来了性能的下降。如果某个字段的内容发生改变，则其位置很可能就需要移动，这会导致碎片的产生。随着数据集中的碎片增加，数据访问性能就会相应降低。这个问题有两种修复方法：

- ☑ 尽可能使用静态数据类型。



- ☑ 经常使用 OPTIMIZE TABLE 语句，它会整理表的碎片，同时恢复由于表更新和删除而导致的空间丢失。

（3）MyISAM 压缩型

有时会创建在整个应用程序生命周期中都只读的表。如果是这种情况，就可以使用 myisampack 工具将其转换为 MyISAM 压缩表来减少空间。在给定硬件配置下（例如，快速的处理器和低速的硬盘驱动器），性能的提升将相当显著。

MyISAM 存储引擎的优势在于占用空间小，处理速度快。缺点是不支持事务的完整性和并发性。

InnoDB 已经开发了十余年，遵循 CNU 通用公开许可（GPL）发行。InnoDB 已经被一些重量级因特网公司所采用，如雅虎、Slashdot 和 Google 等，为用户操作非常大的数据库提供了一个强大的解决方案。InnoDB 给 MySQL 的表提供了事务、回滚、崩溃修复能力和多版本并发控制的事务安全。在 MySQL 从 3.23.34a 开始包含 InnoDB 存储引擎。InnoDB 是 MySQL 上第一个提供外键约束的表引擎，而且 InnoDB 对事务处理的能力也是 MySQL 其他存储引擎所无法与之比拟的。下面介绍 InnoDB 存储引擎的特点及其优缺点。

2. InnoDB 存储引擎

InnoDB 存储引擎中支持自动增长列 AUTO_INCREMENT。自动增长列的值不能为空，且值必须唯一。MySQL 中规定自增列必须为主键。在插入值时，如果自动增长列不输入值，则插入的值为自动增长后的值；如果输入的值为 0 或空（NULL），则插入的值也为自动增长后的值；如果插入某个确定的值，且该值在前面没有出现过，则可以直接插入。

InnoDB 存储引擎中支持外键（FOREIGN KEY）。外键所在的表为子表，外键所依赖的表为父表。父表中被子表外键关联的字段必须为主键。当删除、更新父表的某条信息时，子表也必须有相应的改变。InnoDB 存储引擎中，创建的表的表结构存储在.frm 文件中。数据和索引存储在 innodb_data_home_dir 和 innodb_data_file_path 表空间中。

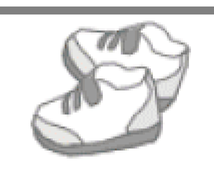
InnoDB 存储引擎的优点在于提供了良好的事务管理、崩溃修复能力和并发控制。其缺点是读写效率稍差，占用的数据空间相对比较大。

InnoDB 表是如下情况的理想引擎：

- ☑ 更新密集的表：InnoDB 存储引擎特别适合处理多重并发的更新请求。
- ☑ 事务：InnoDB 存储引擎是唯一支持事务的标准 MySQL 存储引擎，这是管理敏感数据（如金融信息和用户注册信息）的必需软件。
- ☑ 自动灾难恢复：与其他存储引擎不同，InnoDB 表能够自动从灾难中恢复。虽然 MyISAM 表是能在灾难后修复，但其过程较长。

Oracle 的 InnoDB 存储引擎广泛应用于基于 MySQL 的 Web、电子商务、金融系统、健康护理以及零售应用。因为 InnoDB 可提供高效的 ACID 独立性（Atomicity）、一致性（Consistency）、隔离性（Isolation）、持久性（Durability）兼容事务处理能力，以及独特的高性能和具有可扩展性的构架要素。

另外，InnoDB 设计用于事务处理应用，这些应用需要处理崩溃恢复、参照完整性、高级别的用户并发数，以及响应时间超时服务水平合同。在 MySQL 5.5 中，最显著的增强性能是将 InnoDB 作为默认的存储引擎。在 MyISAM 以及其他表类型依然可用的情况下，用户无须更改配置就可构建基于 InnoDB 的应用程序。





3. MyISAM 和 InnoDB 的区别

（1）InnoDB 不支持 FULLTEXT 类型的索引。

（2）InnoDB 中不保存表的具体行数，也就是说，当执行 select count(*) from table 时，InnoDB 要扫描一遍整个表来计算有多少行，但是 MyISAM 只要简单地读出保存好的行数即可。需要注意的是，当 count(*)语句包含 where 条件时，两种表的操作是相同的。

（3）对于 AUTO_INCREMENT 类型的字段，InnoDB 中必须包含只有该字段的索引，但是在 MyISAM 表中，可以和其他字段一起建立联合索引。

（4）DELETE FROM table 时，InnoDB 不会重新建立表，而是逐行地进行删除。

（5）LOAD TABLE FROM MASTER 操作对 InnoDB 是不起作用的，解决方法是首先把 InnoDB 表改成 MyISAM 表，导入数据后再改成 InnoDB 表，但是对于使用的额外的 InnoDB 特性（如外键）的表不适用。

另外，InnoDB 表的行锁也不是绝对的，如果在执行一个 SQL 语句时 MySQL 不能确定要扫描的范围，InnoDB 表同样会锁全表，如 update table set num=1 where name like “%mr%”。

任何一种表都不是万能的，只用恰当的针对业务类型来选择合适的表类型才能最大地发挥 MySQL 的性能优势。

专家点评

在 MySQL 数据库中，InnoDB 与 MyISAM 数据存储引擎各有特色。这两个表既存在一定相似的联系，也存在着差异。在性能上，可以说是各有千秋，所以用户必须根据自己的业务需要来选择合适的引擎类型，以便提高整个数据库系统的性能，使其达到最大程度的优化。

问题 95　如何在 MySQL 中实现多关键字查询?

问题阐述

在 MySQL 中，查询是最为常用的操作，那么如何实现多关键字的查询呢？

专家解答

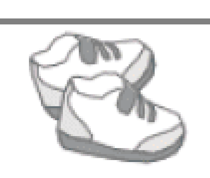
作为刚刚接触 MySQL 数据库的新手，您可能往往会在如查询语句的顺序、排序、语法结构等环节上出现问题。其中，如何应用多关键字查询数据同样也是许多刚刚接触 MySQL 的读者们所困惑的问题之一。实现多关键字查询的方法很简单，其命令结构如下所示：

```
select * from table_name where name like %character1% or …;
```

其中，table_name 是需要查询的数据表名称，“%”作为通配符，用来匹配与两个“%”中间相匹配的字符串内容，or 作为其他关键字查询的连接关键字，一个语句中也可以存在多个“or”连接关键字，用于精确查询数据库表中的数据内容。

专家点评

实现多关键字查询并不难，在查询语句中加入“or”关键字即可，另外，也可以应用多个“and”关键字来精确唯一查询数据，使返回结果更加精确。



问题 96　如何在 MySQL 中修改用户密码和创建权限用户？



Note

问题阐述

在 MySQL 数据库中，如何修改已存在的用户密码？如何新增数据库用户，并赋予该用户特定权限？

专家解答

很多刚刚接触 MySQL 数据库的用户，在默认情况下都会使用系统用户账户，即“root”用户。该用户具有最高管理权限。但是，为了保证数据库数据的安全性和稳定性，需要重置账户密码，当其他用户预申请访问数据库时，就必须通过管理权限用户为访问申请者新增加一个具有特定操作权限的用户账户，以限制用户的操作权限。

1. 修改用户密码

（1）用户可以通过第三方 MySQL 管理工具来修改用户密码，如 phpmyadmin 等。用户只需要登录图形化管理界面，并利用鼠标单击 mysql 数据库中的 user 表，然后再应用 SQL 语句来更改特定用户的账户密码即可。在修改密码时，必须应用 PASSWORD()函数，以此来保证 MySQL 用户名的安全。

（2）在命令提示符中使用 mysqladmin，即在命令提示符界面中输入如下命令：

```
mysqladmin -u root -p oldpassword newpasswd
```

执行上述命令后，需要用户输入 root 的原密码，这样 root 的密码将改为 newpasswd。同样，把命令里的 root 改为用户的用户名，用户就可以更改之前的密码了。当然，如果用户的 mysqladmin 无法正常连接到服务器上，或因软件损坏而导致无法运行，则不能使用该方法，并且 mysqladmin 无法将设置的密码清空。

（3）使用以下语句来修改数据库用户密码，其命令如下：

```
INSERT INTO mysql.user (Host,User,Password) VALUES ('%', 'mr', PASSWORD('mrsoft'));
FLUSH PRIVILEGES
```

此方法更确切地说是用户向数据库中增加一个新用户，用户名为 mr，密码为 mrsoft。注意要使用 PASSWORD 函数，然后还要使用 FLUSH PRIVILEGES 来执行确认。

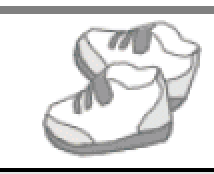
2. 设置用户权限

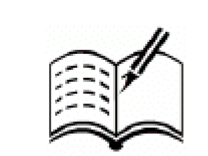
设置用户权限应用的是 Grant 命令。其具体语句如下：

```
GRANT ALL PRIVILEGES ON *.* TO mr@localhost IDENTIFIED BY 'mrsoft' WITH GRANT OPTION;
```

对用户的每一项权限进行设置，插入：

```
mysql>INSERT INTO user VALUES('localhost', '
system',PASSWORD('mrsoft'), 'Y', 'Y', 'Y', '
Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', '
Y', 'Y', 'Y');
```





说明：

设置用户权限的用户账号，必须具备有与“root”相同权限，即所有执行权限。

专家点评

“root”用户是 MySQL 安装时自带的用户，也可以说是权限最大的用户，但是，出于对数据库安全性的考虑，在安装完数据库后建议您修改密码，另外，通过创建其他账户，分配其他用户具有查询、或单一操作的权限（只具有增加、删除或更新权限的用户账户），从而保证数据库系统的安全性和稳定性。

问题 97　在 MySQL 中两种索引 Hash 与 B-Tree 的区别是什么？

问题阐述

在 MySQL 数据库中，Hash 型索引与 B-Tree 型索引有什么区别呢？

专家解答

MySQL 支持 Hash 索引和 B-Tree 索引，但两者究竟有什么区别呢？这个问题可能会困扰许多用户。下面对两者的区别进行详细分析。

MySQL Hash 索引结构具有一定特殊性，其检索效率非常高，索引的检索可以一次定位，不像 B-Tree 索引需要从根节点到枝节点，通过多次 I/O 访问，最后才能访问到页节点，所以 Hash 索引的查询效率要远高于 B-Tree 索引。

但是 Hash 索引的效率要比 B-Tree 高很多，为什么在实际应用中不能都应用 Hash 索引而还要使用 B-Tree 索引呢？任何事物都是有两面性的，Hash 索引也一样，虽然 Hash 索引效率高，但是 Hash 索引由于本身的特殊性也带来了很多限制和弊端，主要有以下几个方面：

（1）MySQL Hash 索引仅仅能满足"=","IN"和"<=>"查询，不能使用范围查询。

由于 MySQL Hash 索引比较的是进行 Hash 运算之后的 Hash 值，所以它只能用于等值的过滤，不能用于基于范围的过滤，因为经过相应的 Hash 算法处理之后的 Hash 值的大小关系，并不能保证和 Hash 运算前完全一样。

（2）MySQL Hash 索引无法被用来避免数据的排序操作。

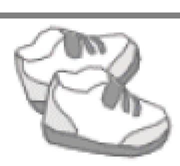
由于 MySQL Hash 索引中存放的是经过 Hash 计算之后的 Hash 值，而且 Hash 值的大小关系并不一定和 Hash 运算前的键值完全一样，所以数据库无法利用索引的数据来避免任何排序运算。

（3）MySQL Hash 索引不能利用部分索引键查询。

对于组合索引，Hash 索引在计算 Hash 值时是组合索引键合并后再一起计算 Hash 值，而不是单独计算 Hash 值，所以通过组合索引的前面一个或几个索引键进行查询时，Hash 索引也无法被利用。

（4）MySQL Hash 索引在任何时候都不能避免表扫描。

上面已经提到，Hash 索引是将索引键通过 Hash 运算之后，将 Hash 运算结果的 Hash 值和所对应的行指针信息存放于一个 Hash 表中，由于不同索引键存在相同的 Hash 值，所以即使取满足某个 Hash 键值的数据的记录条数，也无法从 Hash 索引中直接完成查询，仍要通过访问表中的实际数据进行相应的比较，并得到相应的结果。



（5）MySQL Hash 索引遇到大量 Hash 值相等的情况后性能并不一定就会比 B-Tree 索引高。

对于选择性比较低的索引键，如果创建 Hash 索引，那么将会存在大量记录指针信息存于同一个 Hash 值相关联。这样要定位某一条记录时就会非常麻烦，会浪费多次表数据的访问，从而造成整体性能低下。



Note

问题 98　应用 PHP 操作 MySQL 数据库时应该注意哪些问题？

问题阐述

在使用 PHP 操作 MySQL 数据库时，需要注意的问题都有哪些？

专家解答

众所周知，PHP 和 MySQL 是一对“黄金搭档”，二者具有良好的兼容性和开源性。但是，在使用 PHP 操作 MySQL 数据库时，仍需要注意一些问题。

1. 命令结束使用分号(;)

在使用 MySQL 时，作为命令结束的标志，每一行命令都需要用分号(;)作为结束符。但是应用 PHP 操作 MySQL 数据库时，建议用户把操作结束符分号省略掉，例如：

```
mysql_query ("INSERT INTO tablename (name, sex) VALUES ('$ name', $sex')");
```

这是因为 PHP 也是以分号作为一行的结束标识，额外的分号有时会让 PHP 的语法分析器无法判断哪里是 PHP 语句的结束符，哪里是 MySQL 语句的结束符，所以这里建议用户省略 MySQL 结束符。在这种情况下，虽然省略了分号，但是在应用 PHP 执行 MySQL 命令时，PHP 解释器会自动向语句结尾添加一个结束符标记。

2. TEXT、DATE、和 SET 数据类型

MySQL 数据表的字段必是 MySQL 支持的数据类型。其中以下几个数据类型需要读者特别注意。

TEXT 不是一种数据类型，虽然可能有些书上是这么说的，它实际上应该是“LONG VARCHAR”或“MEDIUMTEXT”类型。

DATE 数据类型的格式是 YYYY-MM-DD，例如，1999-12-08。读者可以很容易地用 date 函数来得到这种格式的当前系统时间：date("Y-m-d")，并且，在 DATA 数据类型之间可以作减法，得到相差的时间天数，其代码如下所示：

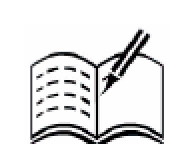
```
$age = ($current_date - $birthdate);
```

集合 SET 是一个有用的数据类型，它和枚举 ENUM 有些相似，只不过 SET 能够保存多个值，而 ENUM 只能保存一个值而已。另外，SET 类型最多只能够有 64 个预定的值，而 ENUM 类型却能够处理最多 65535 个预定义的值。如果需要有大于 64 个值的集合，该怎么办呢？这时就需要定义多个集合来一起解决这个问题了。

3. 通配符

SQL 的通配符有两种，分别是“*”和“%”，分别用于不同的情况。例如，如果用户想看





到数据库的所有内容，可以按如下操作进行查询：

```
SELECT * FROM dbname WHERE USER_ID LIKE '%';
```

从上述语句中可以看出，两个通配符都被用到。它们表示相同的意思，“*”与“%”都是用来匹配任何的字符串，但是它们用在不同的上下文中。“*”用来匹配字段名，而“%”用来匹配字段值。此时还有一个不容易引起注意的地方是，“%”通配符需要和 LIKE 关键字一起使用。

另外，还有一个通配符，就是下划线“_”，它代表的意思和上面不同，是用来匹配任何单个字符的。

4. NOT NULL 和空记录

在应用 PHP 设计带表单和提交功能的页面时，如果用户在没有填任何东西的情况下提交表单内容会如何呢？如果用户确实需要一个值，那么可以用客户端脚本或服务器端脚本来进行数据验证，这一点在前面已经说过了。但是，在数据库中却是允许一些字段被空出来什么也不填。对此类记录，MySQL 将要为之执行一些事情：插入值 NULL，这是默认的操作。

如果用户在字段定义中为之声明 NOT NULL（在建立或修改这个字段时），MySQL 将把这个字段空出来什么东西也不填。对于一个 ENUM 枚举类型的字段，如果为之声明 NOT NULL，则 MySQL 将把枚举集的第一个值插入到字段中。也就是说，MySQL 把枚举集的第一个值作为这个枚举类型的默认值。

一个值为 NULL 的记录和一个空记录是有一些区别的。%通配符可以匹配空记录，但是却不能匹配 NULL 记录。在某些时候，这种区别会造成一些意想不到的后果。建议读者在定义字段时，任何字段都应该声明为 NOT NULL。

如果用户在加入或修改一个新的字段之前，数据库中已经有了一些记录，这时新加入的字段在原来的记录中的值可能是 NULL，也可能为空。这也算是 MySQL 的一个 Bug，所以在这种情况下，使用 SELECT 查询要特别小心。

注意：

在搜索 NULL 时必须用 IS 关键字，而使用 LIKE 关键字时是不会正常发挥其功效的。

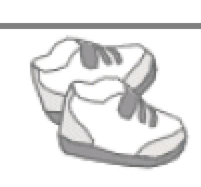
专家点评

应用 PHP 操作 MySQL 数据库需要注意的问题不仅仅局限于上述 4 点，在常用的操作中，还存在诸多需要注意的问题，如向数据库中插入中文产生乱码问题、循环返回数据内容等，这些都有可能成为用户开发过程中的绊脚石。面对一些经常会遇到的问题，用户应该学会在实际应用中积累必要经验，以免在日后的开发工作中因遇到繁琐的数据库操作问题而无从下手。

问题 99　应用 PHP 向 MySQL 插入数据时，如何避免产生乱码？

问题阐述

在应用 PHP 函数 mysql_query()向 MySQL 数据库的特定表中插入数据库时，如何避免在插入数据时产生乱码？





专家解答

这类问题对于开发经验不足的 PHP 程序员来说是比较常见的问题，这是由于编码格式不统一造成的。一般情况下，数据库默认的编码类型为“UTF-8”，如果用户编写的 PHP 文件编码格式为“GBK”或“GB2312”类型，可能造成数据的编码格式不统一，当用户使用其他编码格式的 PHP 文件，向数据库插入文字内容（一般为汉字），应用如 phpMyAdmin 等第三方软件查看数据时，会在该数据库中显示乱码。如果更改数据库的编码格式，则可能会造成前台页面显示不正常。为了解决这一问题，用户可以统一数据库与前台页面的编码格式。

（1）当用户按照原来的方式通过 PHP 存取 MySQL 数据库时，即使设置表的默认字符集为 UTF-8 并且通过 UTF-8 编码发送查询，此时会发现存入数据库的仍然是乱码。问题就出在 connection 连接层上。解决方法是在发送查询指令前执行一下如下命令：

```
SET NAMES 'utf8';
```

这句命令语句相当于下面的 3 句指令：

```
SET character_set_client = utf8;
SET character_set_results = utf8;
SET character_set_connection = utf8;
```

（2）应用第三方软件设置 MySQL 编码格式，这里以 phpMyAdmin 为例。首先，打开 phpMyAdmin 软件。选择指定数据库中的数据表，单击表“结构”选择修改。在“整理”一栏的下拉菜单中选择匹配的编码格式，其结果如图 4.1 所示。

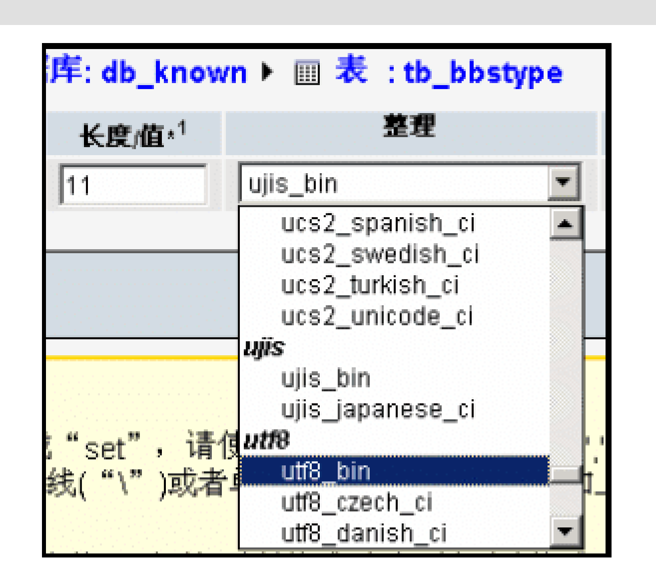


图 4.1　应用 phpMyAdmin 修改数据表编码格式

然后在 PHP 处理页的操作数据库语句后面加入如下语句：

```
mysql_query("set names utf-8");
```

这样，在应用 PHP 向 MySQL 数据库添加内容，就不会因编码不统一而导致出现乱码。

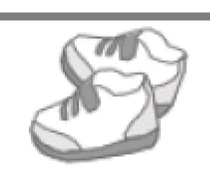
专家点评

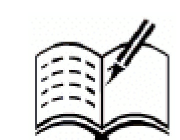
虽然 MySQL 与 PHP 被称作“黄金组合”，但是，对于编码格式的统一问题，读者还是需要多加注意，因为一些不能预期的错误往往都发生在编码格式的交互上。所以，即便是经验丰富的程序员，在开发过程中也应该提高警惕，避免因数据库编码格式和处理页编码格式不统一而造成程序错误。

问题 100　MySQL 常见错误代码有哪些？

问题阐述

在操作 MySQL 过程中，由于操作不当或输入命令的格式不当等原因，经常会出现一些错误或警告，常见的错误代码有哪些？分别代表的错误信息是什么？





专家解答

MySQL 返回的错误代码可以提示用户错误类型，并帮助用户改正错误，用户可以通过常用的错误代码判断错误原因。其中常用的错误代码和相关解释如下所示：

130：文件格式不正确，可以尝试使用 repair.php 来修复，如果不行就只能恢复相关的数据了（此时不是很清楚错误的状况）。

145：文件无法打开，使用 repair.php 修复。

1005：创建表失败。

1006：创建数据库失败。

1007：数据库已存在，创建数据库失败。

1008：数据库不存在，删除数据库失败。

1009：不能删除数据库文件而导致删除数据库失败。

1010：不能删除数据目录而导致删除数据库失败。

1011：删除数据库文件失败。

1012：不能读取系统表中的记录。

1016：文件无法打开，使用后台修复或者使用 phpMyAdmin 进行修复。

1017：服务器非法关机，导致该文件损坏。

1020：记录已被其他用户修改。

1021：硬盘剩余空间不足，需加大硬盘可用空间。

1022：关键字重复，更改记录失败。

1023：关闭时发生错误。

1024：读文件错误。

1025：更改名字时发生错误。

1026：写文件错误。

1030：可能是服务器不稳定（具体原因不是很清楚）。

1032：记录不存在。

1036：数据表是只读的，不能对它进行修改。

1037：系统内存不足，需重启数据库或重启服务器。

1038：用于排序的内存不足，需增大排序缓冲区。

1040：已到达数据库的最大连接数，需加大数据库可用连接数。

1041：系统内存不足。

1042：无效的主机名。

1043：无效连接。

1044：数据库用户权限不足，需联系空间商解决。

1045：数据库服务器/数据库用户名/数据库名/数据库密码错误，需联系空间商检查账户。

1046：没有选择数据库。

1048：字段不能为空。

1049：数据库不存在。

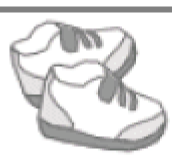
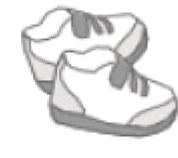
1050：数据表已存在。

1051：数据表不存在。

1054：字段不存在或程序文件与数据库有冲突，需使用正确的程序文件上传并覆盖。

1060：字段重复，导致无法插入这个字段。

1062：字段值重复，入库失败。


1064：MySQL 不支持错误提示中的编码。
1065：无效的 SQL 语句，SQL 语句为空。
1081：不能建立 Socket 连接。
1114：数据表已满，不能容纳任何记录。
1115：设置的字符集在 MySQL 并没有支持。
1116：打开的数据表太多。
1129：数据库出现异常，需重启数据库。
1130：连接数据库失败，没有连接数据库的权限。
1133：数据库用户不存在。
1135：可能是内存不足够，需联系空间商解决。
1141：当前用户无权访问数据库。
1142：当前用户无权访问数据表。
1143：当前用户无权访问数据表中的字段。
1146：数据表缺失，需恢复备份数据。
1147：未定义用户对数据表的访问权限。
1149：SQL 语句语法错误。
1169：字段值重复，更新记录失败。
1177：打开数据表失败。
1180：提交事务失败。
1181：回滚事务失败。
1193：不支持字符集限定(SET NAMES)。
1203：当前用户和数据库建立的连接已到达数据库的最大连接数，需增大可用的数据库连接数或重启数据库。
1205：加锁超时。
1211：当前用户没有创建用户的权限。
1267：不合法的混合字符集。
2002：服务器端口不正确，需咨询空间商以获取正确的端口。
2003：MySQL 服务未启动，需启动该服务。
2013：远程连接数据库有时会出现这个问题，是 MySQL 服务器在执行一条 SQL 语句时失去了连接造成的。

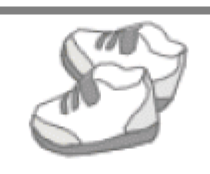
专家点评

常用的错误代码信息可以返回系统提示的错误类型，同时，该错误代码也可以帮助用户对不同的错误或异常及时做出相应处理。了解数据库的常用错误代码，对于程序开发者或数据库管理员来说是非常重要的，这样可以使开发和维护人员更快、更准确地找出错误原因。

问题 101　在 MySQL 中，如何实现多表查询？

问题阐述

为了方便查看多个数据表内容，如何通过 SQL 语句实现多表查询？





专家解答

在实际项目开发过程中，经常需要将不同的信息存储在不同的表中，表与表之间通过某字段互相联系，从而使表的指针形成一种联动关系，从而可以通过 SQL 语言的 SELECT 语句实现多表查询。在 SQL 语言中，通过 where 子句实现多表查询，所要查找的字段名最好以“表名.字段名”格式表示，这样可以防止因表之间字段重名而造成无法获知字段属于哪个表。在 where 子句中多个表之间所形成的联动关系应按“表 1.字段=表 2.字段 and 其他查询条件”形式书写。综上所述，实现多表查询应按如下形式书写：

```
SELECT 字段名 from 表 1,表 2 ... where 表 1. 字段 , 表 2.字段 and 其他查询条件;
```

另外，为了提高 SQL 语句的可读性，经常采取为表取别名的方式对表进行查询。使用表别名可以通过两种方式为表指定别名。

在 MySQL 中，可以通过关键字 as 指定返回查询结果的表别名，其语法结构如下：

```
SELECT 字段名 from 表名 as 表别名 where 条件...;
```

应用 as 关键字为一般形式，其实现过程就像将两张表上的数据表插入第三张表中，并将返回的数据组合成表展现给用户。

在表名后直接加表的别名从而指定表的别名。其语法结构如下：

```
SELECT 字段名 from 表名 表别名 where 条件...;
```

多表查询的示例如下所示：

```
SELECT 字段名 from students as stu, studentinfo as stu_info where stu.id=stuinfo.sid;
```

另外，在 SQL 语言中，使用 union 或 all 关键字可以将多张表的信息合并输出。其中 union 关键字可以将多个 select 语句的查询结果合并输出，并且可以删除重复行，而使用 all 关键字可以将多个 select 语句的查询结果合并输出，但不删除重复行。

注意：

别名通常是一个缩短的表名，用于在连接中引用表中的列名，如果连接中的多个表中有相同的名称列存在，则要求必须使用表名或表的别名限定列名。另外，如果定义了表的别名就不能使用表名。

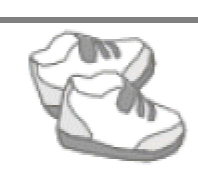
专家点评

在实际数据库查询过程中，往往会查询多个数据表中的多个字段，或者在查询过程中形成内联或相互制约条件，逐条地查询不仅费时费力，在查询过程中也不直观。通过多表查询，可以向用户展示用户所需要查看的列表及数据，从而避免不必要的麻烦。

问题 102　在 MySQL 中，如何实现内连接查询？

问题阐述

在实际开发过程中，经常需要从多个表中查询信息，在 MySQL 数据库中，如何通过内连接实现多表查询呢？





专家解答

内连接查询大多数都比较复杂，如对多张表通过内连接方式建立关联后，还需要在 SQL 语句中再指定一些查询条件，从而进一步实现查询的精确定位。内连接查询的基本语法如下：

```
selectfieldlist from table1[inner] join table2 on table.1column1=table2.column2;
```

其中 fieldlist 为要查询的字段列表，table1 和 table2 为要连接的表名，inner 表示表之间的连接方式为内连接，table1.column1=table2.column2 用于指明表 table1 和 table2 之间的连接条件。

一般情况下，复杂的内连接查询是在基本内连接查询的基础上再附加一些查询条件，例如，下面语句就是一个比较复杂的内连接语句。其代码如下：

```
select tb_yg,.userid,tb_yg.name,tb_yg.sex,tb_yg.bm from tb_yg inner join tb_yg_info on tb_yg.userid=
tb_yg_info.ygid where tb_yg.userid='yg001';
```

上述语句中通过 where 子句指定要查询的员工编号，最终实现内连接表的查询。

专家点评

在实际应用中，通过内连接实现多表查询应用非常广泛。为了简化查询步骤，用户在实际应用中可以使用内连接查询数据。一般情况下，不但可以应用 where 子句，也可以应用如 having 子句、order by 子句实现内连接查询。

问题 103　在 MySQL 中，如何实现外连接查询?

问题阐述

为了从多个表中查询符合条件信息，在 MySQL 数据库中，如何通过外连接实现部分或全部匹配结果集。

专家解答

内连接返回的是两个或多个表中符合条件的数据，而外连接返回部分或全部匹配行，这主要取决于所建立的外连接的类型。在实际应用中常采用"outer join"关键字实现外连接，其中要包含连接的两个表中的不匹配行，可以通过左连接和右连接方式实现。

应用外连接的一般查询语法结构如下：

```
select * from tablename1 left/right outer join tablename2 on tablename1. column1=tablename2. column2
```

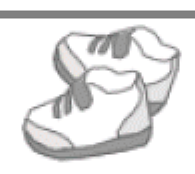
tablename1、tablename2 是需要连接的数据表，left/right 用于指定该外连接是应用左连接或右连接方式将数据表 tablename1 和数据表 tablename2 进行连接。

上述结构中，提到应用左连接或有连接方式实现两个表中的不匹配行。下面以左连接为例，向读者展示如何应用左连接返回查询结果。

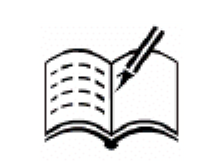
左连接返回的查询结果包含坐标中的所有符合查询条件及右表中所有满足连接条件的行，MySQL 数据库中使用左连接的语法格式如下：

```
select field1[field2…] from table1 left [outer] join table on join_condition[where search_condition];
```

其中 lefit outer join 表示表之间通过做连接方式相互连接，也可以简写成 left join；on



join_condition 值多表建立连接所使用的连接条件；where search_condition 为可选项，用于设置查询条件。



专家点评

与左连接(left outer join)类似，右连接可以将右表中满足查询条件的记录和坐标中满足连接条件的记录全部显示出来。对于刚刚接触 MySQL 数据库的用户来说，这可能会稍微花费他们的一点时间来了解这些内容。应用外连接同样可以实现连接匹配并拼接两个数据表中的相关字段，这样可以使用户在查询过程中，更清晰地查看所需要访问的数据内容。

问题 104　在 MySQL 中，如何实现嵌套查询？

问题阐述

嵌套查询在数据库系统开发过程中应用广泛，如何在 MySQL 中利用嵌套查询，来提高复杂的查询语句可读性呢？

专家解答

嵌套查询在数据库系统开发中可以对复杂查询进行分解，从而提高查询语句的可读性。其中复杂查询可以分为简单的嵌套查询和复杂的嵌套查询。

下面通过一个简单的示例向读者介绍如何应用 SQL 语句的嵌套，来实现嵌套查询。子查询是一个 select 语句，返回单个值且嵌套在 select、insert、update 和 delete 语句中。任何可以使用表达式的地方都可以应用子查询作为嵌套查询条件。其中子查询也称作内部查询或内部连接，包含子查询的语句成为外部查询或外连接，这些都可以应用到嵌套查询中。简单的嵌套查询语句如下：

```
select * from tablename1 where sid in(select sid from tablename2 where tid=num);
```

其中 tablename1 和 tablename2 分别是被查询的数据表；“sid”与“tid”作为子查询限定条件。

另外，在实际的数据库系统开发中，对两张表实现嵌套相对容易，对多表实现嵌套查询相对麻烦一些。一般情况下，可以连续在查询语句的子句中嵌套 select 语句。其基本的形式如下所示：

```
select * from tablename1 where sid in (select sid from tablename2 where name in (select…));
```

另外，实现多表的嵌套时，可以同时使用谓词，如 any、some、all，这些谓词可以和比较运算符一起使用，判断是否任何或全部返回值都满足搜索条件。其中 some 和 any 是存在量的，判断返回值是否满足搜索条件，这两个词含义相同，可以替换使用；all 谓词称为通用谓词，它只关心是否有谓词都满足搜索条件。其中谓词的语法形式如下：

```
scalar_expression{查询条件运算符}{some|any|}(subquery)
```

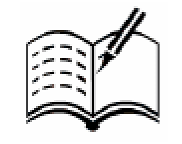
其中 scalar_expression 指任何有效的 SQL 表达式；subquery 指包含某列结果集的子查询。

专家点评

嵌套查询在实际项目开发过程中应用比较繁琐，它可以通过一条 SQL 语句实现多表之间



的复杂查询，与子查询不同的是，子查询中只是通过本数据表的相关字段内容来精确查询条件，而嵌套查询可能会从其他数据表中获取其他字段名称进行查询。另外，嵌套查询中可能也包含子查询条件。与连接查询不同的是，用户需要查询的是某个表的数据，而不是将几个数据表连接后再显示的数据，所以在实际应用过程中一定要灵活运用嵌套查询。



Note

问题 105　如何连接远程 MySQL 服务器？

问题阐述

在实际的生产环境中，我们所架设的 PHP 程序运行环境，多数情况下 Apache 服务器与 MySQL 服务器是不放置在同一台服务器中的，这样就会涉及一个问题，即如何实现程序与 MySQL 服务器的连接？如果不能够很好地实现与 MySQL 服务器的连接，其他任何操作都无从谈起。

专家解答

在连接远程 MySQL 服务器之前，首先，应该为远程服务器创建一个远程登录用户，然后，以远程登录用户的身份登录服务器，为将要访问这个远程 MySQL 服务器的 IP 地址开通一个用户，最后就可以通过这个用户来访问远程的服务器了。

例如，MySQL 服务器安装于 IP 为 192.168.1.194 的机器上，项目存储于 192.168.1.59 的机器上，下面就在 192.168.1.59 的机器上连接 192.168.1.194 机器上的 MySQL 服务器。其具体操作步骤如下：

（1）在 192.168.1.194 的机器上，以 localhost 服务器、root 用户的身份登录 MySQL 服务器，为 MySQL 服务器添加一个用户，设置用户名为 root，服务器为 192.168.1.194，密码为 111，其拥有所有的权限。在 phpMyAdmin 图形化管理工具中添加用户的操作如图 4.2 所示。

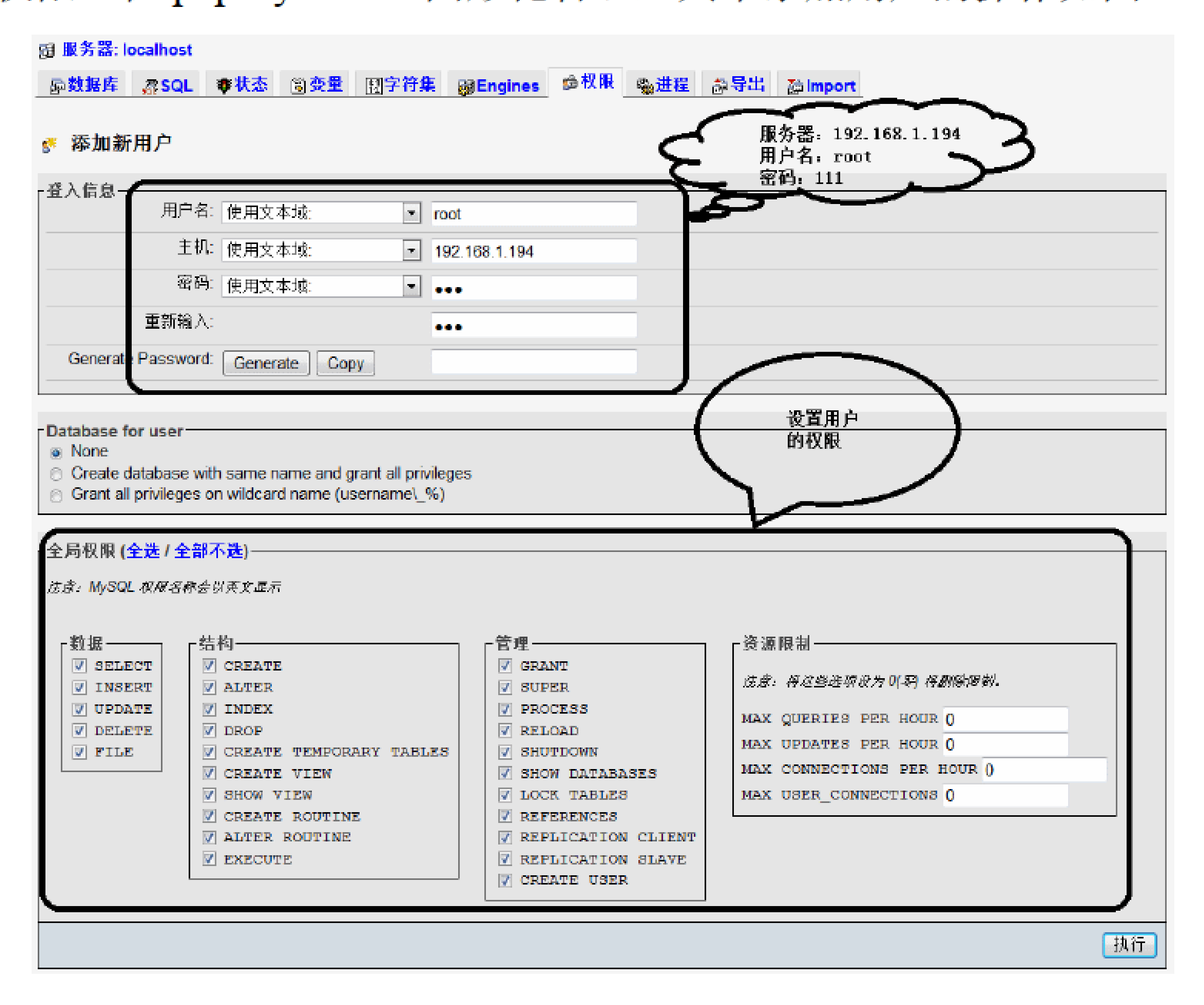
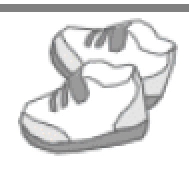
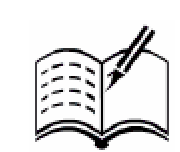


图 4.2　添加数据库用户（一）





（2）在 192.168.1.194 的机器上，以新添加的用户（192.168.1.194 服务器、root 用户、密码 111）身份登录 MySQL 服务器，并为该服务器添加一个新用户，同时设置用户名为 mr，服务器为 192.168.1.59，密码为 mrsoft，其拥有数据的添加、修改、删除和查询的权限。在 phpMyAdmin 图形化管理工具中添加用户的操作如图 4.3 所示。

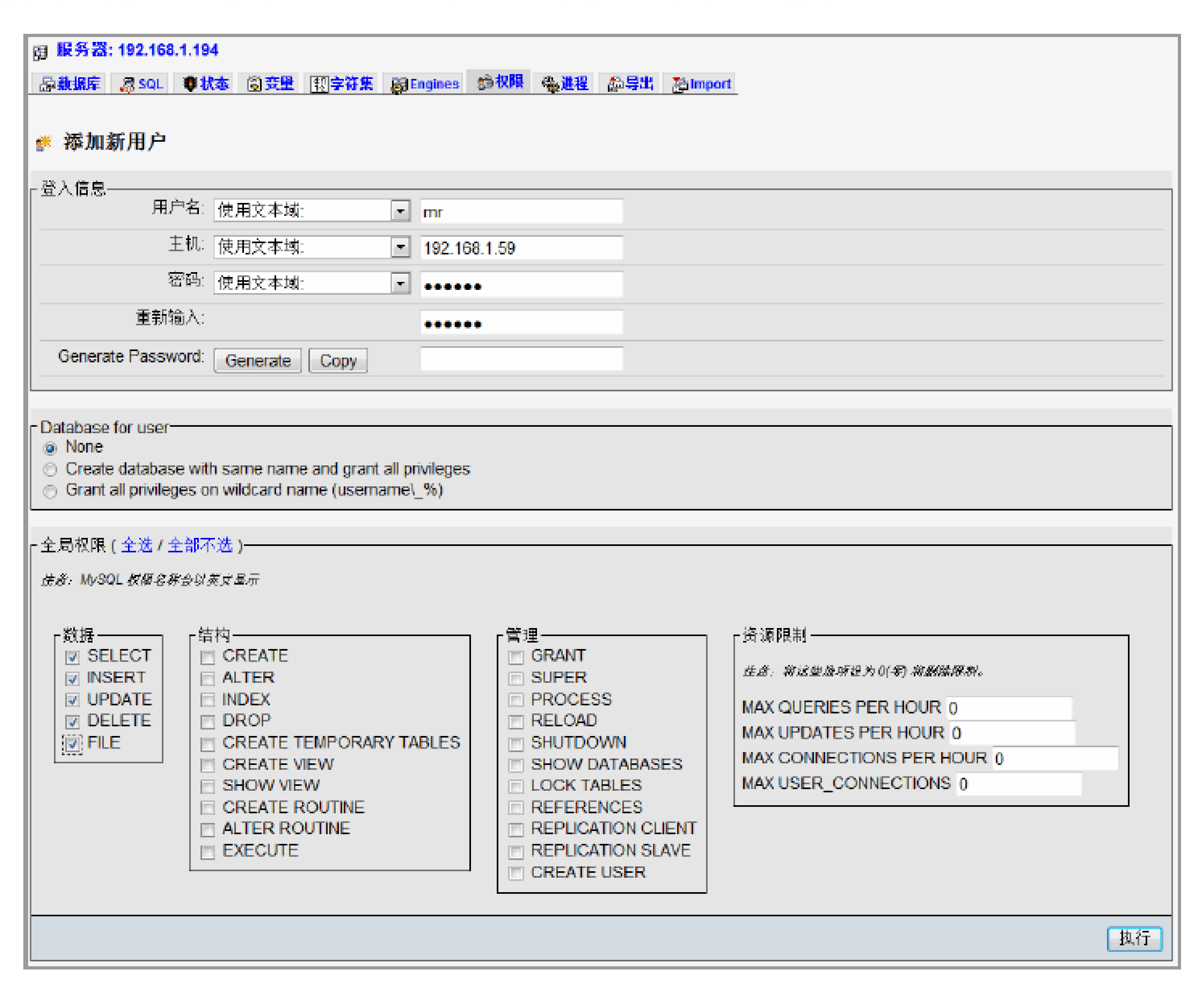


图 4.3　添加数据库用户（二）

（3）在 192.168.1.59 的机器上，在 PHP 项目中定义连接远程 MySQL 服务器的用户名为 mr，密码为 mrsoft，设置服务器为 192.168.1.59，成功连接远程 MySQL 服务器。其代码如下：

```
<?php
$conn=mysql_connect("192.168.1.194","mr","mrsoft") or die(mysql_error());
?>
```

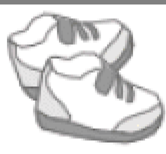
注意：

在连接远程的 MySQL 服务器时必须关闭 MySQL 服务器所在机器的防火墙，否则 MySQL 服务器是不允许远程连接的。

专家点评

在实际的生产环境中，项目和数据库的分隔存储是一个非常好的方案，不但能够提高其安全性，而且还可以降低服务器被攻击后的损失程度。上述介绍的是通过 phpMyAdmin 图形化管理工具来创建远程 MySQL 服务器用户，在通过 phpMyAdmin 图形化管理工具管理 MySQL 数据库时有一项非常重要的设置，即登录时所输入的用户名和密码，在 phpMyAdmin 图形化管理工具的配置文件（config.inc.php）中默认设置的是 localhost 服务器，用户名为 root，密码为空。如果用户想要通过 192.168.1.194 的服务器，以 root 用户、密码 111 的身份进行登录，那么必须对配置文件（config.inc.php）进行修改，修改内容如图 4.4 所示。

在完成上述修改之后，即能够以 192.168.1.194 服务器，用户名 root，密码 111 的身份登录到 MySQL 服务器。需要注意的是，只有以这个身份登录，为指定的 IP 地址以及对应的用户授权后，这个对应的用户才具备远程访问的权限。

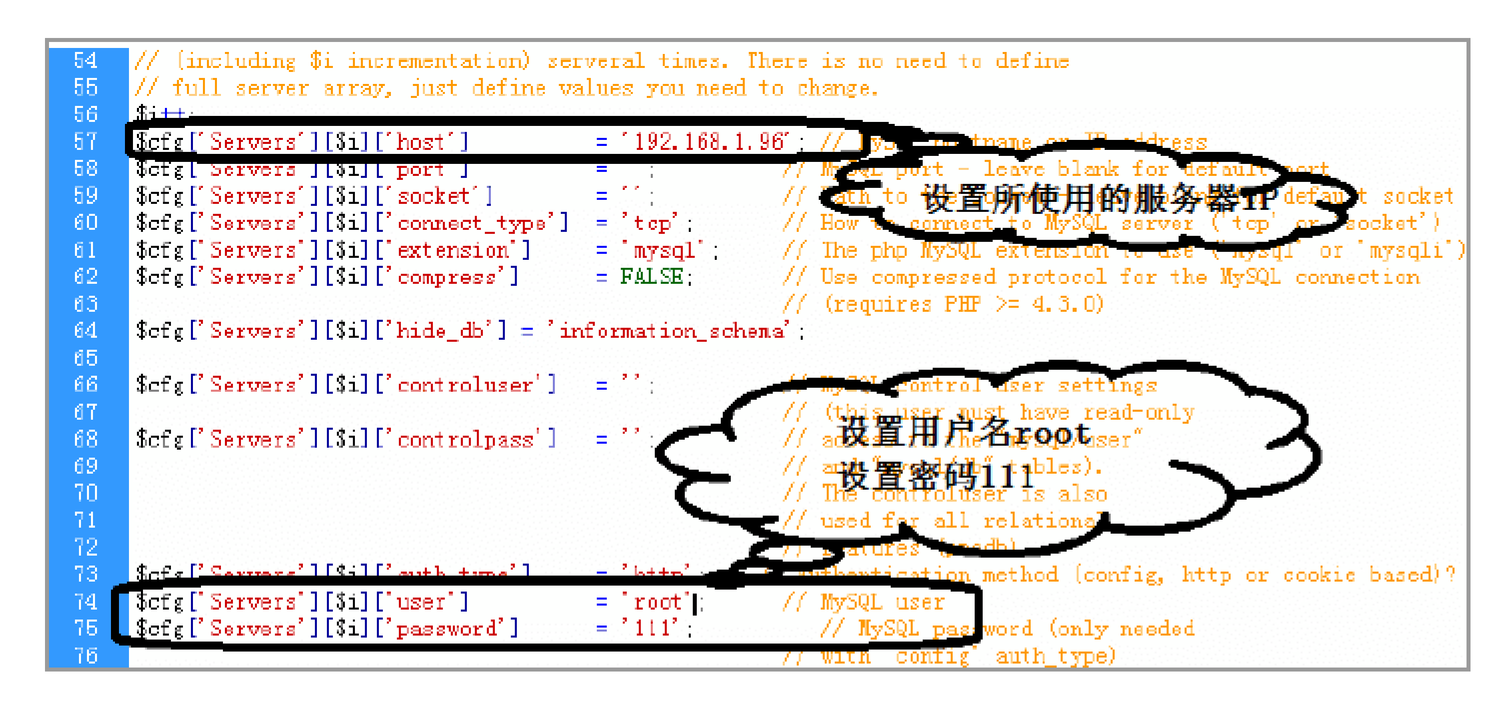


图 4.4　修改配置文件（config.inc.php）

问题 106　如何实现 MySQL 数据库的双机热备份？

问题阐述

在 Web 项目中其最薄弱的环节就是它的服务器，一旦服务器受到攻击，就将面临内容被删除、数据被篡改的危险，更有甚者会导致服务器瘫痪。为了保证数据的安全，对数据进行备份是必要的。备份数据最佳的方法就是实现数据库的双机备份，即存在两个数据库服务器，当程序执行一次操作时，同时将数据保存到两个数据库服务器中，从而确保数据的安全性，即使有一个数据库服务器瘫痪，那么还有另外一个数据库服务器可以使用。我们这里所要解决的问题就是如何实现 MySQL 数据库的双机热备份。

专家解答

双机热备份通过日志文件来传输服务器上数据的变化。主服务器进程在数据被更新时触发，并将相应的日志文件发送到从服务器，从服务器进程接收到主服务器发送的日志文件后，阅读日志文件并对响应的数据库进行操作。具体的工作流程如图 4.5 所示。

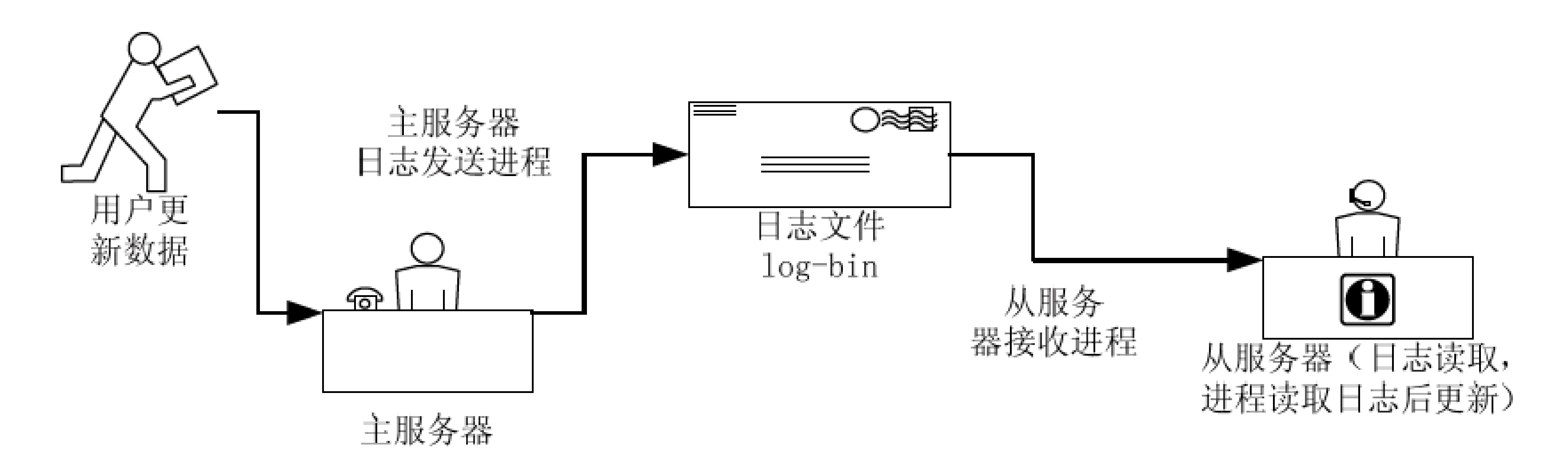


图 4.5　双机热备份的工作原理

1. 主服务器的配置

主服务器的 IP 设置为 192.168.1.194。配置主服务器的步骤如下：

（1）找到 MySQL 数据库的配置文件 my.ini。该文件可能存储于 Windows 的安装目录或 MySQL 的安装目录之下。

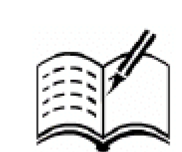
（2）通过编辑器打开 my.ini 文件，定位到[mysqld]一节，并在该节中加入如下内容：

```
server-id=1
log-bin=c:/log-bin.log
binlog-do-db=dbname
```



其中 server-id 指定服务器编号；log-bin 为日志的存储位置及文件名；binlog-do-db 为进行日志记录的数据库名称，如果存在多个数据库，则用逗号“，”分隔。

例如，指定服务器编号为 1，日志的存储位置和名称是 C 盘根目录下 log-bin.log，指定对 db_backup 数据库进行日志记录操作。



（3）创建 db_backup 数据库。

（4）进入 MySQL 的命令操作界面，通过命令为从服务器授权访问数据库的用户名和密码。命令如下：

```
grant replication slave on *.* to 'mr'@'192.168.1.96' identified by 'mrsoft' ;
```

其中指定用户名是 mr，从服务器的 IP 是 192.168.1.96，密码是 mrsoft。命令执行的操作界面如图 4.6 所示。

（5）重启 MySQL 服务器，使更改生效。

（6）重新进入 MySQL 的命令操作界面，执行如下命令检测配置是否生效。

```
show master status \G
```

若出现如图 4.7 所示的结果，则说明配置成功。

```
C:\WINDOWS\system32\cmd.exe - mysql -uroot -h192.168.1.194 -p111
Microsoft Windows [版本 5.2.3790]
(C) 版权所有 1985-2003 Microsoft Corp.

C:\Documents and Settings\Administrator>mysql -uroot -h192.168.1.194 -p111
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 20
Server version: 5.0.51b-community-nt-log MySQL Community Edition (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> grant replication slave on *.* to 'mr'@'192.168.1.96' identified by 'mrso
ft';
Query OK, 0 rows affected (0.00 sec)

mysql> _
```

图 4.6　执行为从服务器授权的操作

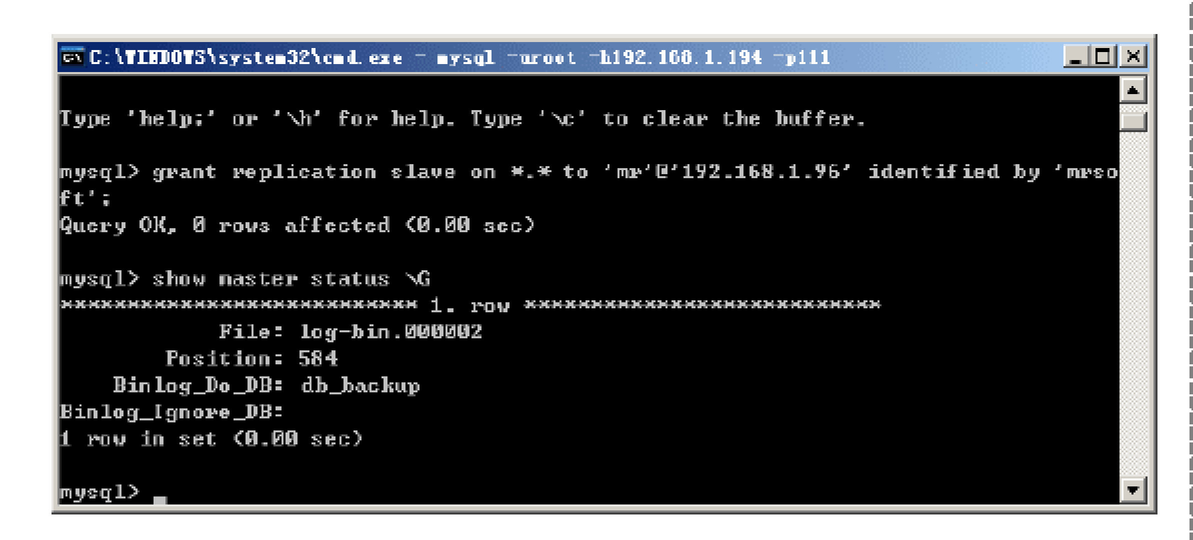


图 4.7　检测 MySQL 主服务器配置

2. 从服务器的配置

（1）在从服务器中创建与主服务器中相同的数据库。

（2）找到从服务器中的 my.ini 文件，定位到[mysqld]一节，并在该节中加入如下内容：

```
server-id=2
master-host=192.168.1.194
master-port=3306
master-user=mr
master-password=mrsoft
master-connect-retry=60
```

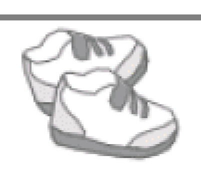
其中 server-id 指定从服务器的编号，该编号必须与主服务器不同；master-host 指定主服务器的 IP 地址；master-port 指定访问主服务器所用端口号；master-user 和 master-password 指定访问主服务器的用户名和密码；master-connect-retry 表示如果因为网络问题与主服务器断开连接，则等待多少秒尝试重新连接。

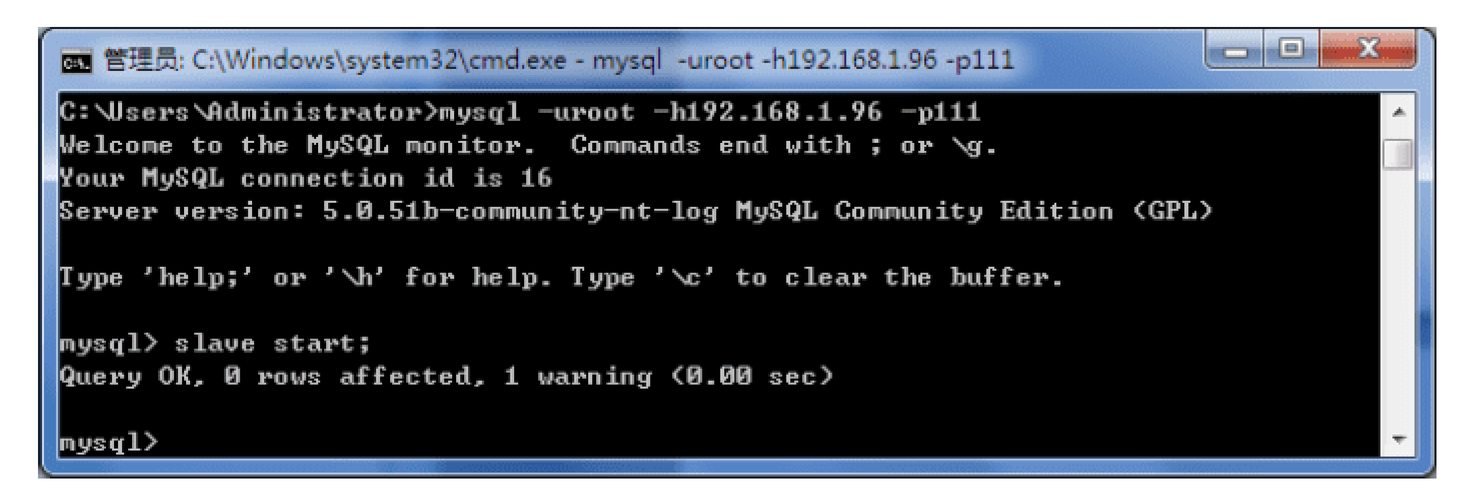
（3）重新启动 MySQL 服务器，使配置生效。

（4）进入从服务器的 MySQL 命令操作界面，执行如下命令启动进程。

```
slave start;
```

命令执行的界面如图 4.8 所示。





```
管理员: C:\Windows\system32\cmd.exe - mysql -uroot -h192.168.1.96 -p111
C:\Users\Administrator>mysql -uroot -h192.168.1.96 -p111
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 5.0.51b-community-nt-log MySQL Community Edition (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> slave start;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql>
```

图 4.8　从服务器启动进程

至此，MySQL 数据库双机热备份配置完成。

专家点评

双机热备份是一种很好的备份数据库的方法，在应用该方法时，有一点必须要注意，就是必须关闭从服务器的防火墙，否则主服务器中的 MySQL 数据是不能添加到从服务器的。

问题 107　解决 MySQL 语句错误的最佳方案是什么？

问题阐述

在应用 PHP 内置函数操作 MySQL 数据库时，MySQL 语句产生的错误是很难发现的，因为在 PHP 脚本中执行 MySQL 的添加、修改、查询和删除语句时，它不会输出任何的信息，除非对该语句是否执行成功进行了判断。这也就导致无法判断出 SQL 语句是否被正确执行，无法捕捉到 MySQL 语句的错误信息。这就是我们在这里将要解决的问题。

专家解答

在讲解解决方案之前，我们有必要先来了解一下错误的根源，正所谓“对症下药，方可药到病除”。PHP 操作 MySQL 数据库出现错误的地方可以归结为 3 个环节：第一是数据表单的提交，第二是数据的处理，第三是存储到数据库。这 3 个环节是 MySQL 最容易产生错误的地方。

第一个环节表单提交页和数据处理页之间，在表单提交页中输入的文本域名称一定要与提交到数据处理页的变量相一致，且大小写必须统一，因为获取表单提交数据的$_GET 和$_POST 方法是区分大小写的。如果不一致，那么文本域的值是不会被存储到数据库中的。

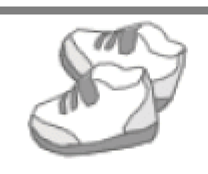
第二个环节是数据处理页，在该页的 SQL 语句中字段名称排列的顺序与提交的变量值的顺序要一一对应。如果顺序不对应，由于字段的类型不同，将导致数据不能正常添加。

第三个环节是数据处理页和数据表之间，在数据处理页编写的 SQL 语句中使用的字段名称一定要与数据表中的字段名称相同。

了解了问题的根源后，下面看解决方案。如何才能准确地捕捉到这个错误信息，最佳的方案就是应用 mysql_error()函数返回错误信息。该函数在 mysql_query()函数后应用，能够准确地获取 SQL 语句在执行过程中出现的错误，如果没有错误则不输出任何内容。

1. 错误举例

首先建立一个数据库“test”，在该数据库下建立一个数据表“user”。该数据表的结构如表 4.1 所示。



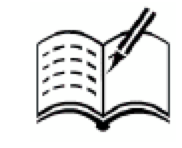


表 4.1　数据表 user 结构

字　段　名	类　　型	主　　键	说　　明
id	Int(10)	是	自动编号
user	Varchar(30)		用户名
pass	Varchar(30)		密码
lxdh	int(10)		联系电话
lxdz	Varchar(30)		联系地址

然后建立与数据库的连接，connect.php 的代码如下：

```
<?php
$id=mysql_connect('localhost','root','root');
mysql_select_db('test',$id );
?>
```

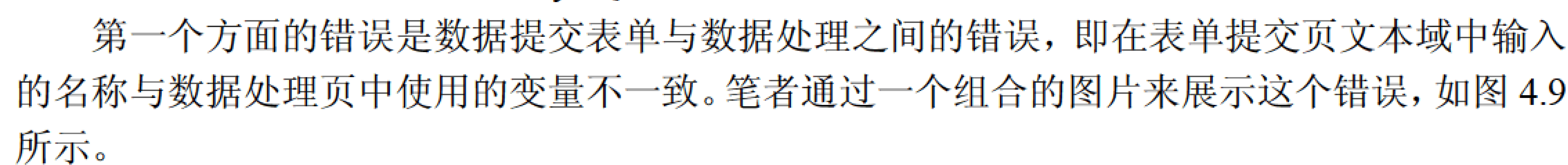
连接数据库成功后将建立一个表单提交的页面 submit.php，代码如下：

```
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
<title>错误调试</title></head>
<body>
<table width="300" border="1" align="center" cellpadding="0" cellspacing="0">
 <form name="form1" method="post" action="submit_ok.php">
  <tr>
    <td width="94" height="30" align="center" valign="middle">用户名:</td>
    <td width="200"><input name="user" type="text" id="user" size="20"></td>   </tr>
  <tr>
    <td height="31" align="center" valign="middle">密码:</td>
    <td><input name="pass" type="text" id="pass" size="20"></td>   </tr>
  <tr>
    <td height="28" align="center" valign="middle">联系电话:</td>
    <td><input name="lxdh" type="text" id="lxdh" size="20"></td>   </tr>
  <tr>
    <td height="30" align="center" valign="middle">联系地址:</td>
    <td><input name="lxdz" type="text" id="lxdz" size="20"></td></tr>
  <tr>
    <td height="30" colspan="2" align="center" valign="middle"><input name="submit" type="submit"
id="submit" value="提交"></td> </tr>
  </form></table>
</body></html>
```

进行数据处理的文件是 submit.php，代码如下：

```
<?php
       include("connect.php");                    //连接到数据库
       if($submit=="提交"){
          $query="insert into user (user,pass,lxdh,lxdz) values('$user','$pass','$lxdh','$lxdz')";
          $result=mysql_query($query,$id);
          if($result){
             echo "<meta http-equiv=\"Refresh\" content=\"2;url=$furl\">";
             echo "发布成功!";
          }else{ echo "发布主题失败!!"; }
```

```
            }
    ?>
```



通过上述 3 个文件，使用 MySQL 语句的错误基本就可以展示出来了。

第一个方面的错误是数据提交表单与数据处理之间的错误，即在表单提交页文本域中输入的名称与数据处理页中使用的变量不一致。笔者通过一个组合的图片来展示这个错误，如图 4.9 所示。

从图 4.9 中可以看到，在设置用户名的文本字段时，文本域中填写的是“usee”，而在数据处理页中，通过提交使用的变量则是“$user”，这就是错误所在。这个错误在程序运行的过程中是看不出来的，程序仍然正常运行，但是当进入到数据库中查看数据时就会出现如图 4.10 所示的结果。

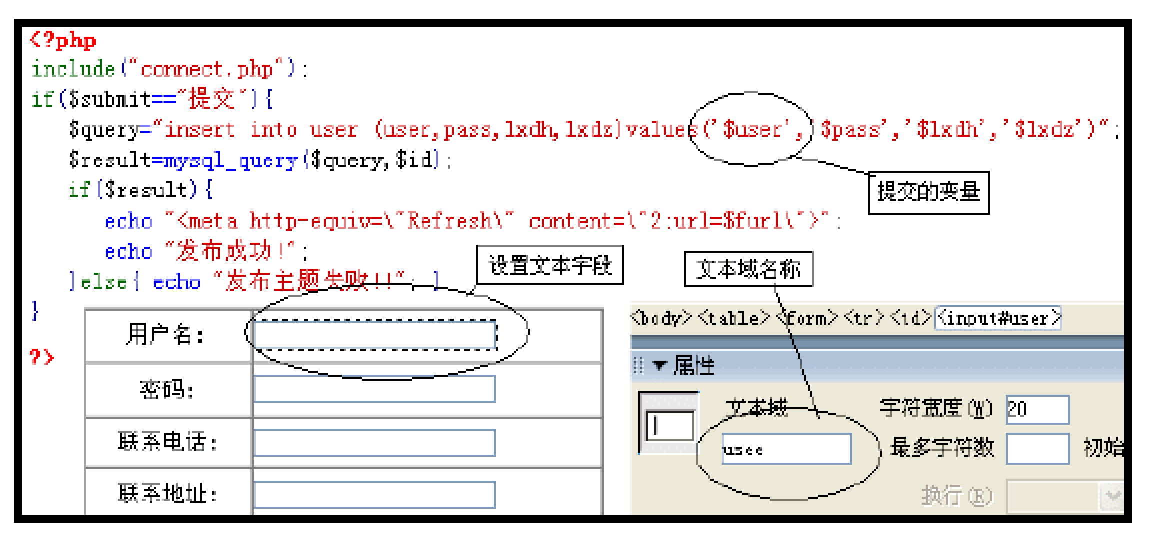


图 4.9　错误示例（一）



id	user	pass	lxdh	lxdz
2	"飞许"	"123456"	7785792	"吉林"
3	"汉将"	"456789"	5202588	"松原"
1	"流云"	"123456"	7680305	"长春"
4	"桃李"	"456123"	1335033	"四平"
6		"123456"	7680305	"白城"

图 4.10　数据表中的数据

其中“id”为 6 的记录中的“user”字段的值是空的，其原因就是由于文本域名称和提交的变量不一致。

第二个方面的错误是在数据处理页中，在该页的 SQL 语句中字段名称排列的顺序与提交的变量值的顺序不一致，如图 4.11 所示。

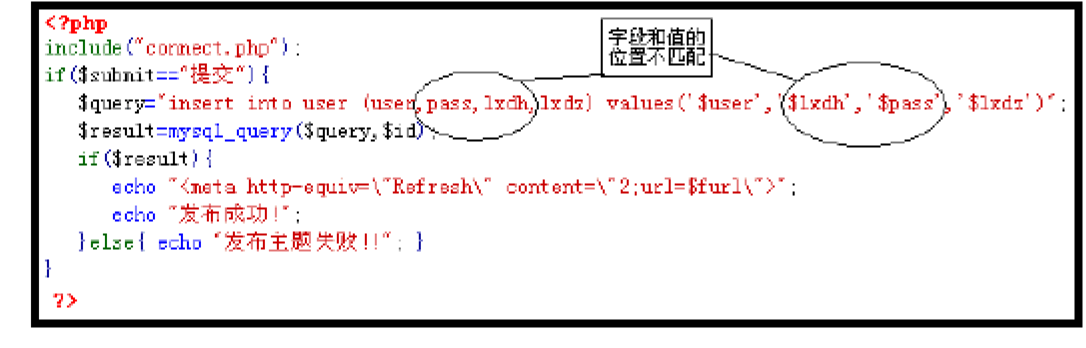


图 4.11　错误示例（二）

如图 4.11 所示，在 insert 语句中的字段的顺序是“user,pass,lxdh,lxdz”，而字段的值的顺序是“'$user','$lxdh','$pass','$lxdz'”，其中颠倒了“lxdh”和“pass”的顺序，该程序仍然可以继续运行，但是在查看数据库时问题就显示出来了，如图 4.12 所示。

从数据库中可以看到，第 7 条记录的“pass”和“lxdh”字段的值被颠倒了，“lxdh”字段的值被添加到“pass”中了，另外，由于“lxdh”字段的类型是“int”，不支持“varchar”类型的值，导致“lxdh”字段中没有值。

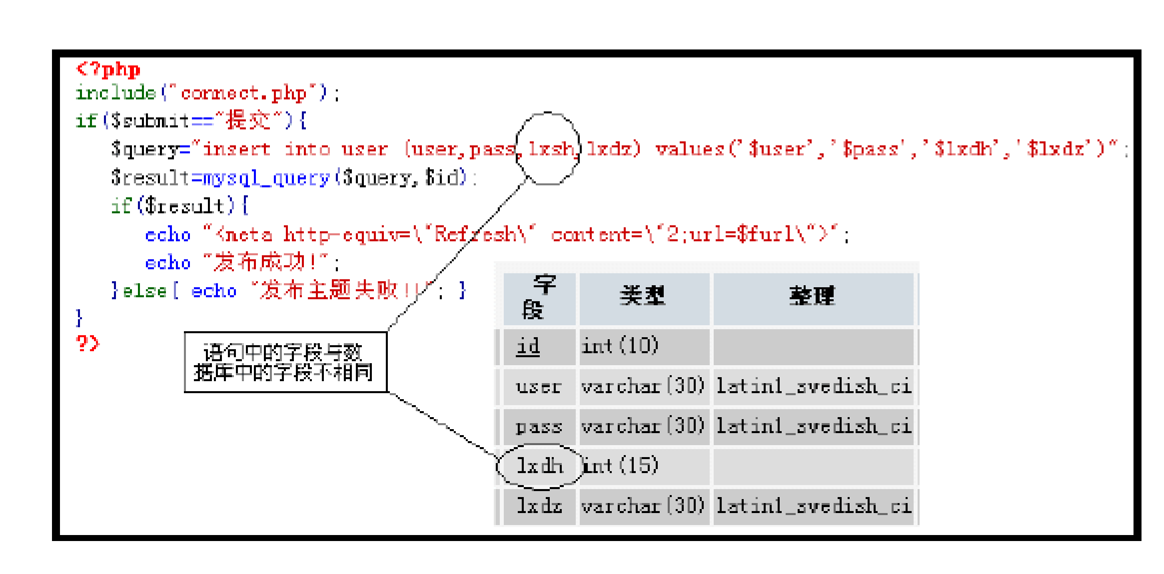
第三个方面的错误是 insert 语句中的字段名和数据库中的字段名称不一致，如图 4.13 所示。

其中的字段“lxsh”和数据表中的字段名称“lxdh”不一致，这个错误直接导致数据提交失败。



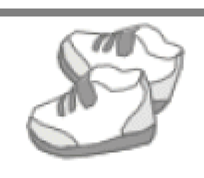
id	user	pass	lxdh	lxdz
2	"飞许"	"123456"	7785792	"吉林"
3	"汉将"	"456789"	5202588	"松原"
1	"流云"	"123456"	7680305	"长春"
4	"桃李"	"456123"	1335033	"四平"
7	"小米"	"7680305"		"白城"

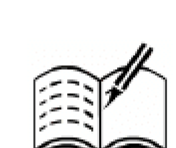
图 4.12



字段	类型	整理
id	int(10)	
user	varchar(30)	latin1_swedish_ci
pass	varchar(30)	latin1_swedish_ci
lxdh	int(15)	
lxdz	varchar(30)	latin1_swedish_ci

图 4.13　错误示例（三）





2. 获取错误信息

使用“echo mysql_error();”来输出错误信息，该语句被放置在数据处理页中使用，具体的位置是在 mysql_query()函数输出的后面，如图 4.14 所示。

通过该语句的使用输出错误信息，如图 4.15 所示。

```
<?php
include("connect.php");
if($submit=="提交"){
   $query="insert into user (user,pass,lxsh,lxdz) values('$user','$pass','$lxdh','$lxdz')";
   $result=mysql_query($query,$id); //送出一个query字符串
   echo mysql_error();
   if($result){
      echo "<meta http-equiv=\"Refresh\" content=\"2;url=$furl\">";
      echo "数据提交成功!";
   }else{ echo "数据提交失败!!"; }
}
?>
```

输出错误信息语句



图 4.14　输出错误信息语句的使用

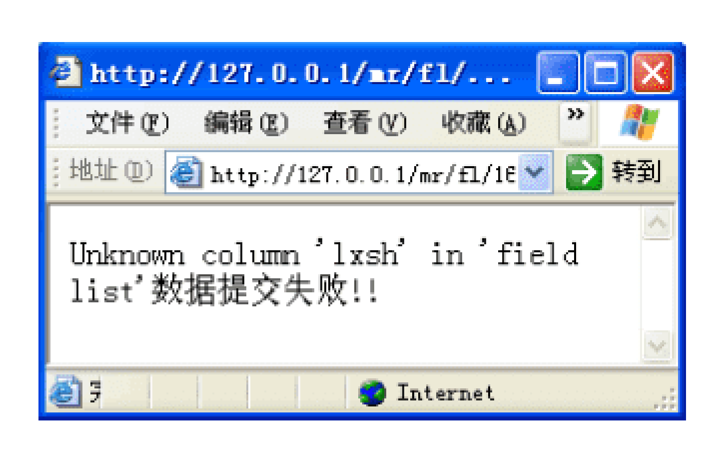


图 4.15　输出错误信息的页面

专家点评

通过 mysql_error()函数获取错误信息，不但可以获取 insert 添加语句的错误信息，而且也适用于 select、update 和 delete 语句，同时还可以在连接数据库服务器及选择数据库时应用此函数，获取连接、选择失败时的错误信息。例如：

```
<?php
$id=mysql_connect('localhost','root','111') or die(mysql_error());
mysql_select_db('test',$id ) or die(mysql_error());
mysql_query("set names gb2312");
?>
```

问题 108　如何执行数据的批量添加?

问题阐述

如何执行数据的批量添加?

专家解答

数据的批量添加使用 LOAD DATA 和 MySQLIMPORT 语句实现。

LOAD DATA 语句，通过读取本地系统上的文件，可以将大量数据添加到数据库中。语法如下：

```
load data local infile "filename.txt" into table table_name
```

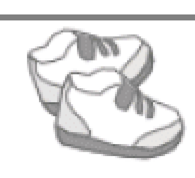
其中，filename.txt 是当前目录中的数据文件的名称，table_name 是要被装载的数据表。

LOAD DATA 语句读取客户机当前目录中数据文件 filename.txt 的内容，将其内容装载到数据表 table_name 中。

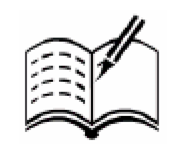
MySQLIMPORT 语句，实用程序直接从文件读取批量数据，它相当于 load data 语句的一个接口。语法如下：

```
%MySQLimport -local table_name filename.txt ;
```

MySQLIMPORT 可以自动生成一个 LOAD DATA 语句，该语句把 filename.txt 文件中的数



据装入 table_name 表中。MySQLIMPORT 根据文件名导出表名，即将文件名第一个圆点前的所有字符作为表名。例如，文件 class.txt 被装入 class 表中。



问题 109　如何规范地定义数据表中的字段名称？

问题阐述

数据表中的字段名称用来代表一组数据的名称，合理地为每组数据定义最贴切的字段名称是很有必要的，那么，如何规范地定义数据表中的字段名称呢？

专家解答

数据表字段的命名必须遵循以下规范：

（1）采用有意义的字段名。字段的名称必须是易于理解，能表达字段功能的英文单词或缩写英文单词，单词首字母必须大写，一般不超过 3 个英文单词。例如，人员信息表中的电话号码可命名为：Telephone 或 Tel。产品明细表中的产品名称可用 ProductName 表示（推荐一般用完整的英文单词）。

（2）系统中所有属于内码字段（仅用于标示唯一性和程序内部用到的标示性字段），名称取为：“ID”，采用整型或长整型数，具体根据可能的数据量确定，增加记录时取最大值加 1，该字段通常为主关键字。

（3）在命名表的列时不要重复表的名称。例如，在名为 Employee 的表中避免使用名为 EmployeeLastName 的字段。

（4）不要在列的名称中包含数据类型。

（5）字段最好填写描述信息。

（6）在定义数据表中字段的名称时，切记不能使用特殊的关键字，如 mysql、time、datetime、database 等。

例如，下面创建 tb_book 表，并创建相应字段，如图 4.16 所示。

专家点评

在创建字段时，最好为字段添加注释，这样可以方便程序维护及调试。



字段	类型	整理	属性	Null	默认	额外	操作
id	int(10)			否		auto_increment	
sort	varchar(100)	utf8_unicode_ci		否			
talk	varchar(100)	utf8_unicode_ci		否			
books	varchar(100)	utf8_unicode_ci		否			
sy 图书名称	varchar(10000)	utf8_unicode_ci		否			
catalog	varchar(10000)	utf8_unicode_ci		否			
bookpath	text	utf8_unicode_ci		否			
programpath	text	utf8_unicode_ci		否			
videopath	text	utf8_unicode_ci		否			
date	date			否			

图 4.16　创建数据表字段



Note

问题 110　如何查询指定时间段的数据？

问题阐述

在查询数据库中的数据时，如何实现查询指定时间段的数据呢？

专家解答

1. 方法分析

在 SQL 语句中实现对指定范围内的数据进行查询可以通过以下两种方式实现，第一种方式是通过关键字 between…and…实现，语法格式如下：

```
select 要查找的字段 from 表名 where 字段名 between 初始值 and 终止值
```

第二种方式是通过比较运算符实现，语法格式如下：

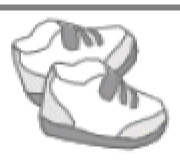
```
select 要查找的字段 from 表名 where 字段名>初始值 and 字段名<终止值
```

2. 应用示例

在图书管理系统中，查询指定出版日期范围内的图书情况。其关键步骤如下：

创建 index.php 文件，启动 MySQL 服务并连接 MySQL 数据库。当单击“查看”按钮时，显示指定时间段的数据。代码如下：

```
<?php
    if($_POST[sub]){                                              //单击查看按钮
        //判断文本框信息
        if($_POST[text1] == "" || $_POST[text1] == "输入开始日期" || $_POST[text] == "" ||
$_POST[text] == "输入结束日期"){
            echo "<script>alert('请输入日期');</script>";          //JavaScript 提示
        }else{
    //验证日期格式的正则表达式
    if(preg_match("/([0-9]{3}[1-9]|[0-9]{2}[1-9][0-9]{1}|[0-9]{1}[1-9][0-9]{2}|[1-9][0-9]{3})-(((0[13578]|1[0-2])-
(0[1-9]|[12][0-9]|3[01]))|((0[469]|11)-(0[1-9]|[12][0-9]|30))|(02-(0[1-9]|[1][0-9]|2[0-8])))/",$_POST[text])){
                //连接 MySQL
                $conn = mysql_connect("localhost","root","111") or die("connect mysql false");
                //连接 MySQL 数据库
                mysql_select_db("db_database",$conn) or die("connect database false");
                mysql_query("set names utf8");                    //设置编码格式
                $rss = mysql_query("select * from tb_demo047 where date between
'$_POST[text1]' and '$_POST[text]' order by date");              //返回结果集
                echo'<table width="580px"><tr><td background="pic/head.JPG">ID</td><td
background="pic/head.JPG">书名</td><td background="pic/head.JPG">价格</td><td background=
"pic/head.JPG">日期</td></tr>';
                while($rst = mysql_fetch_array($rss)){            //循环输出结果
    ?>
        <tr><td background="pic/head.JPG"><?php echo $rst[0];?></td><td background=
"pic/head.JPG"><?php echo $rst[1];?></td><td background="pic/head.JPG"><?php echo $rst[2];?></td><td
background="pic/head.JPG"><?php echo $rst[3];?></td></tr>
    <?php
```




Note

```
                }
            }else{
        //JavaScript 提示信息
                echo "<script>alert('日期格式错误！正确格式为：xxxx-xx-xx');</script>";
            }
        }
    }
?>
```

运行本示例，如图 4.17 所示，首先在图中的文本框中输入日期范围，然后单击“查看”按钮即可实现查找该日期范围内的所有图书信息。



ID	书名	价格	日期
7	PHP24堂课	78元	2011-01-11
8	C#24堂课	78元	2011-01-17
9	JAVA24堂课	78元	2011-01-17
10	C24堂课	78元	2011-01-17

图 4.17　查询指定时间段的数据

问题 111　如何按月查询统计数据？

问题阐述

在查询数据库中的数据时，如何按月查询统计数据呢？

专家解答

SQL 语言中提供了如下函数，利用这些函数可以很方便地实现按年、月、日进行查询。

year(data)：该函数用于返回日期表达式 data 中的公元年份所对应数值。

month(data)：该函数用于返回日期表达式 data 中的月份所对应的数值。

day(data)：该函数用于返回日期表达式 data 中的日期所对应的数值。

例如，按月查询图书信息，其 SQL 语句如下：

```
select * from tb_demo where month(date) = '$_POST[text1]' order by date
```

问题 112　如何实现在删除数据前给予提示信息？

问题阐述

在开发网站的过程中，为了避免将重要的数据删除，有必要在执行删除操作之前提醒一下管理者，是否要删除该数据，这样可以避免误删数据。那么，如何实现在删除数据前给予提示信息呢？



专家解答

1. 解决思路

删除数据前给予提示信息，应用 JavaScript 脚本实现，其代码如下：

```
<script language="javascript">
function delconfirm(x){
if(window.confirm('您确定删除该项吗?')==true){
  window.location.href='index_ok.php?ljid='+x;}}
</script>
```

使用 window 对象中的 confirm()方法实现在确认对话框中显示指定的字符串，并通过 window 对象的 location 属性指定当前文档要跳转到的 URL。

2. 应用示例

本例实现在执行删除操作时弹出一个提示对话框“您确定删除该项吗?”，运行结果如图 4.18 所示。



图 4.18 删除数据前给予提示信息

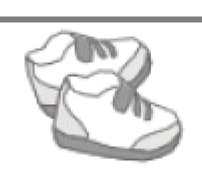
实现步骤如下：

（1）创建与数据库的连接，代码如下：

```
<?php
$id=mysql_connect("localhost","root","root")or dir('连接服务器失败:'. mysql_error());
if(mysql_select_db("db_database",$id)){
echo "";
}else{echo ('连接数据库失败:'. mysql_error());}
mysql_query("set names gb2312");
?>
```

（2）从数据库中读取数据，设置删除操作的按钮，编写 JavaScript 脚本实现在删除数据之前弹出对话框给予提示，关键代码如下：

```
<?php session_start();include("Conn/conn.php");$furl=getenv("HTTP_REFERER")?>
<script language="javascript">
function delconfirm(x){
if(window.confirm('您确定删除该项吗?')==true){
  window.location.href='index_ok.php?ljid='+x;}}
</script>
<?php
  $query="select * from tb_delete where id limit 4";
  $result=mysql_query($query);
  while(@$myrow=mysql_fetch_array($result)){
  ?>
  <tr>
    <td width="77" height="36" align="center"><span class="STYLE3"><?php echo $myrow[user];?>
</span></td>
    <td width="96" align="center"><span class="STYLE3"><?php echo $myrow[title];?></span></td>
    <td width="155" align="center"><span class="STYLE3"><?php echo $myrow[text];?></span></td>
    <td width="95" align="center"><span class="STYLE3"><?php echo $myrow[data];?></span></td>
     <td width="77" align="left"><input name="Submit" type="submit" onClick="delconfirm(<?php
echo $myrow[id];?>)" value="删除"></td>
```



```
</tr>
  <?php }?>
```

（3）执行数据的删除操作，代码如下：



```
<?php session_start();include("Conn/conn.php");$furl=getenv("HTTP_REFERER");
if($ljid==true){
$ljid=$_GET['ljid'];
$query="delete from tb_delete where id='$ljid'";
$result=mysql_query($query);
if($result==true){
  echo "删除成功!!";
  echo "<meta http-equiv=\"Refresh\" content=\"2;url=index.php\">";
}else{echo "删除失败!";}}
?>
```

问题 113　如何查询日期型数据？

问题阐述

如何查询日期型数据？

专家解答

不同的数据库对日期型数据的查询是有区别的，下面将以几种典型的数据库为例讲解在不同的数据库中对日期型数据的查询方式。

下面均是以在图书表（tb_demo）中查询出版日期（date）为“2011-07-19”为例进行讲解的。

1. MySQL 数据库中对日期型数据进行查询

```
select * from tb_demo where birthday='2011-07-19'
```

2. SQL Server 数据库中对日期型数据进行查询

```
select * from tb_demo where birthday='2011-07-19'
```

3. Access 数据库中对日期型数据进行查询

```
select * from tb_demo where birthday=#2011-07-19#
```

通过上面 3 个例子可以发现，在 MySQL 数据库和 SQL Server 数据库中实现对日期型数据查询所要查询的日期应用单引号括起来，而在 Access 数据库中使用 JET SQL 语法查询时所查询的日期应用“#”号括起来。

问题 114　如何使用左连接（left outer join）查询数据？

问题阐述

如果要包含连接的两个表中的不匹配行，可以通过左连接或右连接的方式实现。那么，如何使用左连接查询数据呢？





专家解答

左连接返回的查询结果包含左表中的所有符合查询条件及右表中所有满足连接条件的行，MySQL 数据库中使用左连接的语法格式如下：

```
select field 1[field2…] from table1 left [outer] join table2 on join_condition [where search_condition]
```

参数说明：

❶ left outer join：表示表之间通过左连接方式相互连接，也可以简写成 left join。

❷ on join_condition：指多表建立连接所使用的连接条件。

❸ where search_condition：可选项，用于设置查询条件。

例如，查询员工信息表（tb_demo078_informaction）中所有记录以及员工工资表（tb_demo078_wages）中与员工 ID 相匹配的记录。

首先，进入 phpMyAdmin 中，选择 db_database08 数据库，单击“SQL”按钮，进入输入查询语句页面，如图 4.19 所示。

输入完查询语句后，单击“执行”按钮，即可显示查询结果，如图 4.20 所示。

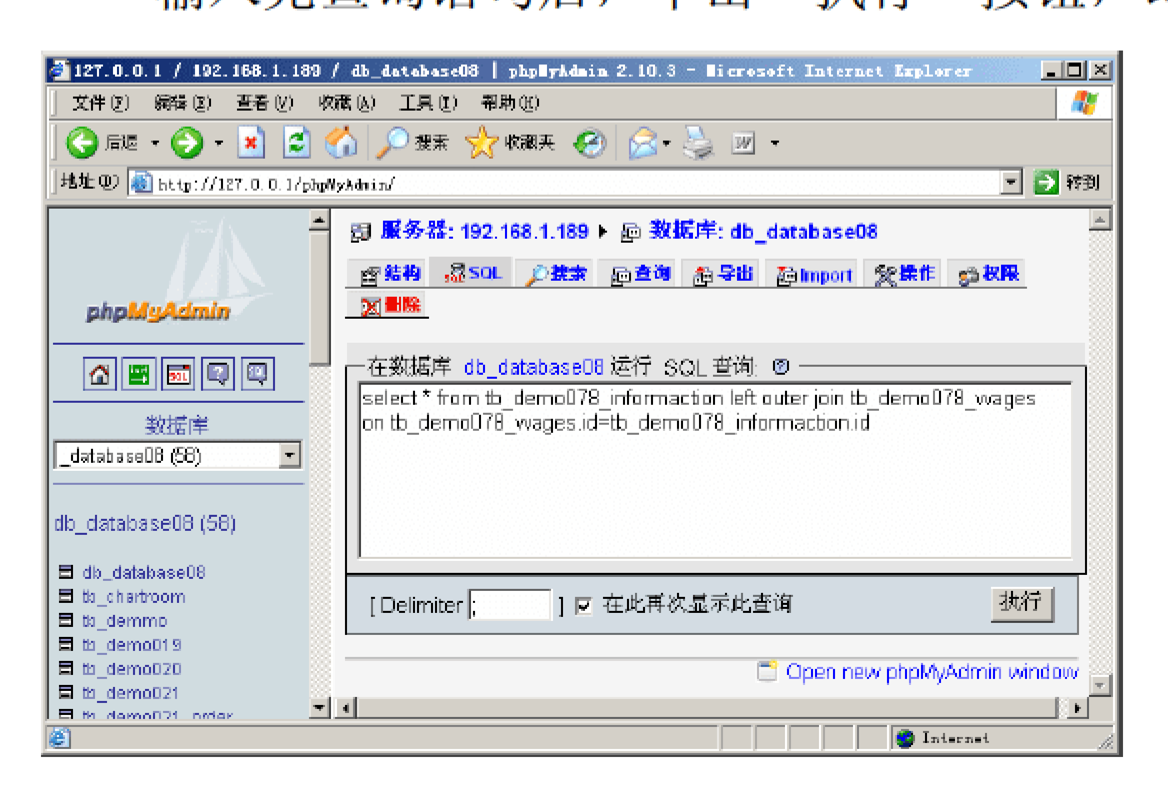


图 4.19　在 phpMyAdmin 中输入查询语句

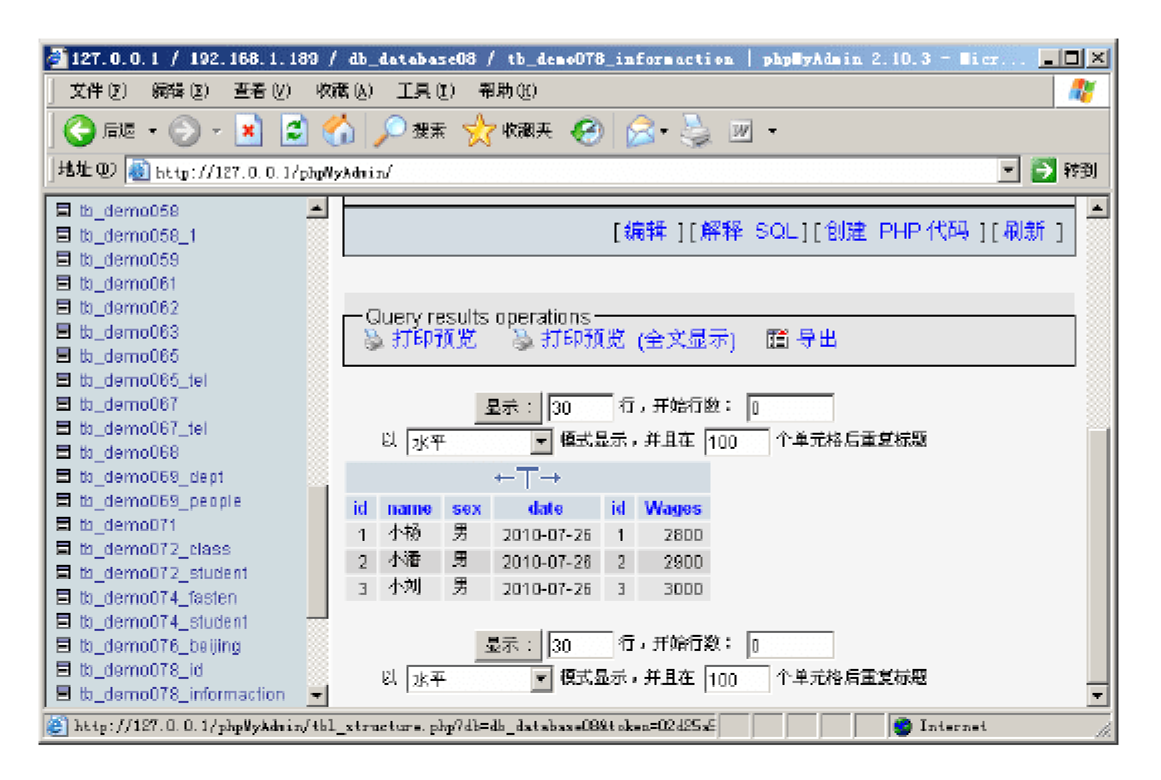


图 4.20　查询结果

专家点评

左连接是以 left outer join 左侧的表名为基准的，故称左连接，左侧表名的所有信息将被全部输出，符合条件的右侧表信息也会被输出，如果不符合条件，则返回为 NULL。

问题 115　如何使用右连接（right outer join）查询数据？

问题阐述

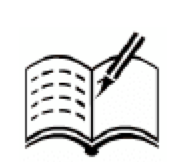
通过右连接可以将右表中满足查询条件的记录全部显示出来并显示左表中满足连接条件的记录。那么，如何使用右连接（right outer join）查询数据呢？

专家解答

右连接返回的查询结果包含左表中的所有符合连接条件以及右表中所有满足查询条件的行，MySQL 数据库中使用右连接的语法格式如下：

```
select field 1[field2…] from table1 right [outer] join table2 on join_condition [where search_condition]
```





参数说明：

❶ right outer join：表示表之间通过右连接方式相互连接，也可以简写成 right join。

❷ on join_condition：指多表建立连接所使用的连接条件。

❸ where search_condition：可选项，用于设置查询条件。

例如，将员工信息表中满足连接条件的记录及员工工资表中所有记录全部显示出来。

首先，进入 phpMyAdmin 中，选择 db_database08 数据库，单击“SQL”按钮，进入输入查询语句页面，如图 4.21 所示。

输入完查询语句后，单击“执行”按钮，即可显示查询结果，如图 4.22 所示。

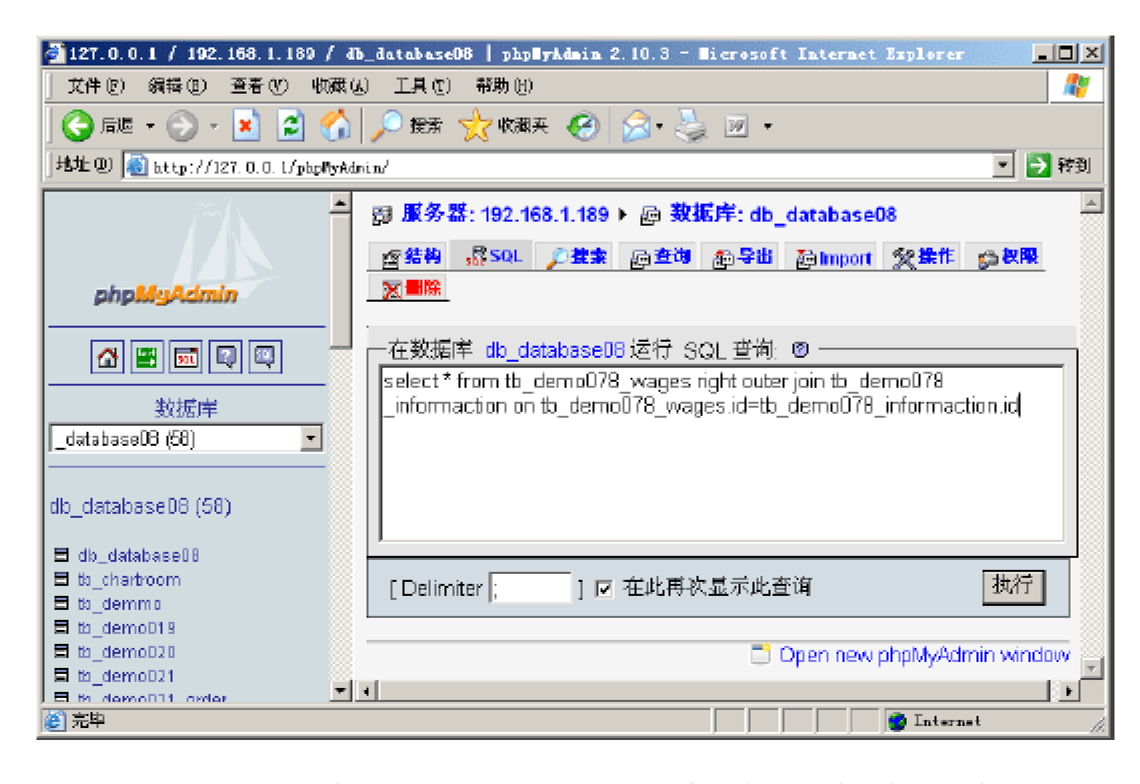




图 4.21　在 phpMyAdmin 中输入查询语句

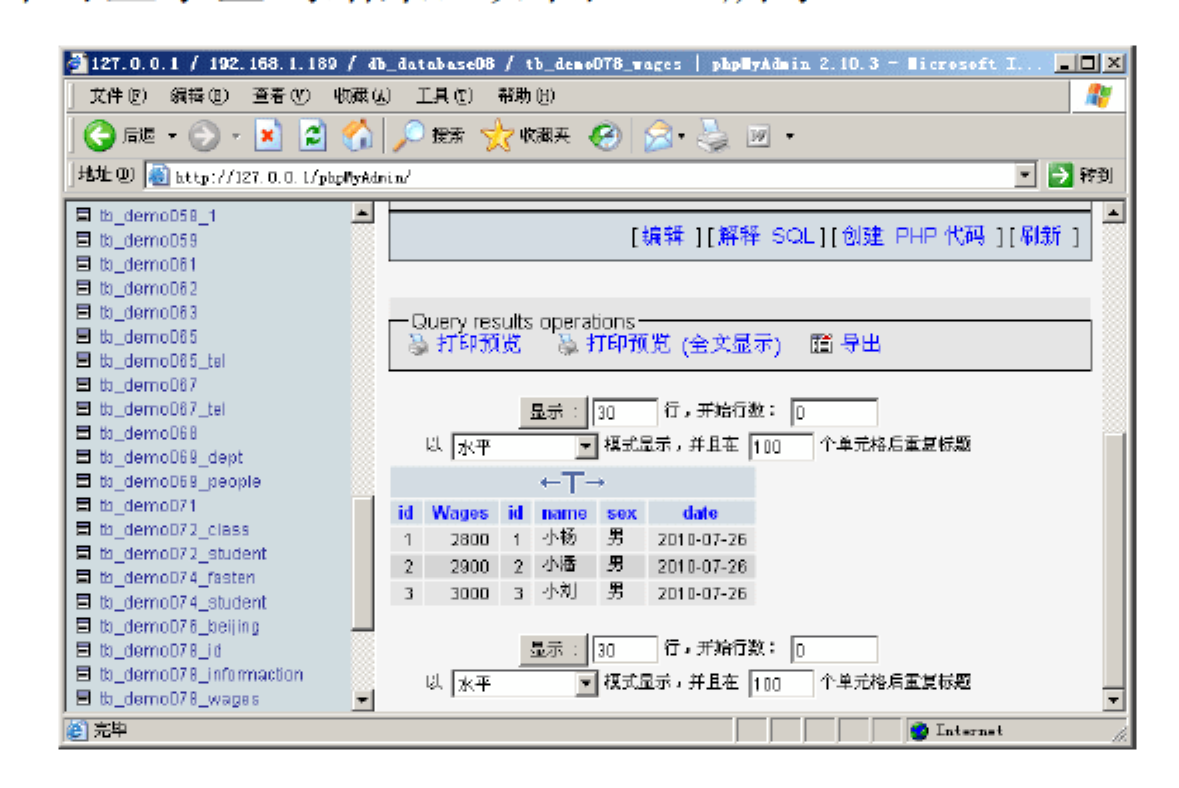


图 4.22　查询结果

专家点评

右连接是以 right outer join 右侧的表名为基准的，故称右连接，右侧表名的所有信息将被全部输出，符合条件的左侧表信息也会被输出，如果不符合条件，则返回为 NULL。

问题 116　如何对统计结果进行排序？

问题阐述

在商品信息管理系统中，经常需要对商品的价格、数量进行求和，然后根据所求得的值对商品进行排序，从而实现商品的排序输出。那么，如何对统计结果进行排序呢？

专家解答

使用 ORDER BY 可以对查询的结果进行升序（ASC）和降序（DESC）排列，在默认情况下，ORDER BY 按升序输出结果。如果要按降序排列，则可以使用 DESC 来实现。语法格式如下：

```
ORDER BY 字段名 [ASC|DESC];
```

ASC 表示按升序进行排序。

DESC 表示按降序进行排序。

说明：

当对含有 NULL 值的列进行排序时，如果是按升序排列，则 NULL 值将出现在最前面；如果是按降序排列，则 NULL 值将出现在最后。





例如，查询 tb_login 表中的所有信息，按照“id”进行降序排列，查询语句如下：

```
select * from tb_login order by id desc;
```

查询结果如图 4.23 所示。

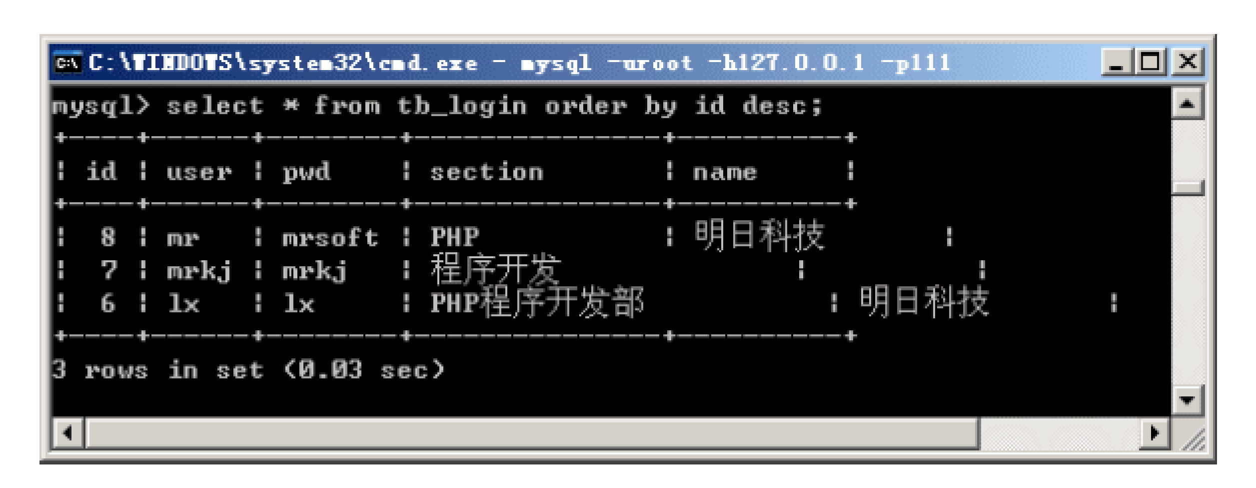


图 4.23　按 id 序号进行降序排列

问题 117　如何使用聚集函数 SUM 对学生成绩进行汇总？

问题阐述

在学生成绩管理系统中，经常需要求出班级所有学生的各科成绩总和，从而求出所有学生的各科平均成绩。那么，如何使用聚集函数 SUM 对学生成绩进行汇总呢？

专家解答

在 SQL 语句中，实现对某字段的所有记录进行求和应用的是 sum()函数，其语法如下：

```
sum([all | distinct] expression)
```

参数说明：

❶ all：表示对指定字段的所有值进行聚集函数运算，all 为默认值，如果不加参数 all 或 distinct 则表示对指定字段的所有记录进行聚集运算。

❷ distinct：表示对指定字段的所有非重复记录进行求和。

❸ expression：是精确数字或近似数字数据类型分类（bit 数据类型除外）的表达式。

sum()函数的返回类型如表 4.2 所示。

表 4.2　sum()函数的返回类型

表达式结果	返 回 类 型
整数分类	int
decimal分类	decimal(38,s) 除以decimal(10,0)
money和smallmoney分类	money
float和real分类	float

例如，下面将学生的各科成绩进行汇总，得出总分。

首先，进入 phpMyAdmin 中，选择 db_database08 数据库，单击“SQL”按钮，进入输入查询语句页面，如图 4.24 所示。

输入完查询语句后，单击“执行”按钮，即可显示查询结果，如图 4.25 所示。





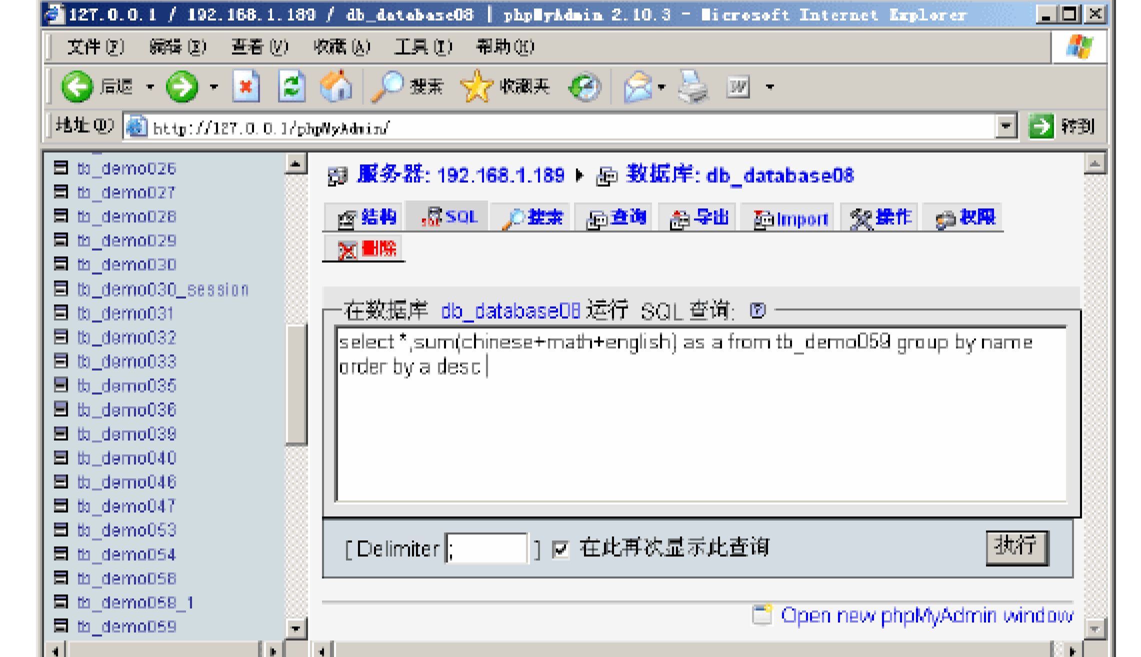


图 4.24　在 phpMyAdmin 中输入查询语句

图 4.25　查询结果

专家点评

sum()函数主要被用于数据表字段求和，一般与分组函数联合使用，但是只有出现在多表查询之中才能真正体现出它的优势。

问题 118　如何使用聚集函数 MIN 求利润最少的商品？

问题阐述

在销售管理系统中，商家会根据不同商品的销售利润来确定该商品的进货数量。那么，如何使用聚集函数 MIN 求利润最少的商品呢？

专家解答

SQL 语言中获取某字段的最小值应用 min()函数，函数的语法如下：

```
min([all | distinct] expression)
```

例如，查询出销售利润最小的商品信息。

首先，进入 phpMyAdmin 中，选择 db_database08 数据库，单击“SQL”按钮，进入输入查询语句页面，如图 4.26 所示。

输入完查询语句后，单击“执行”按钮，即可显示查询结果，如图 4.27 所示。

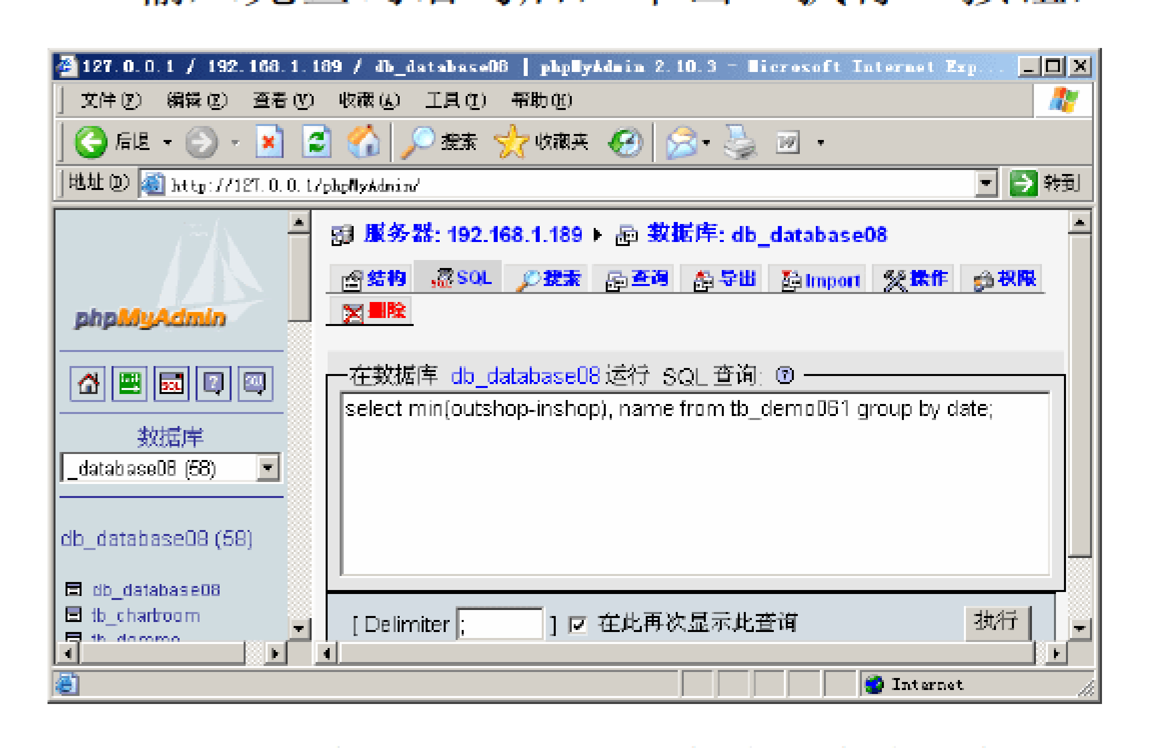


图 4.26　在 phpMyAdmin 中输入查询语句

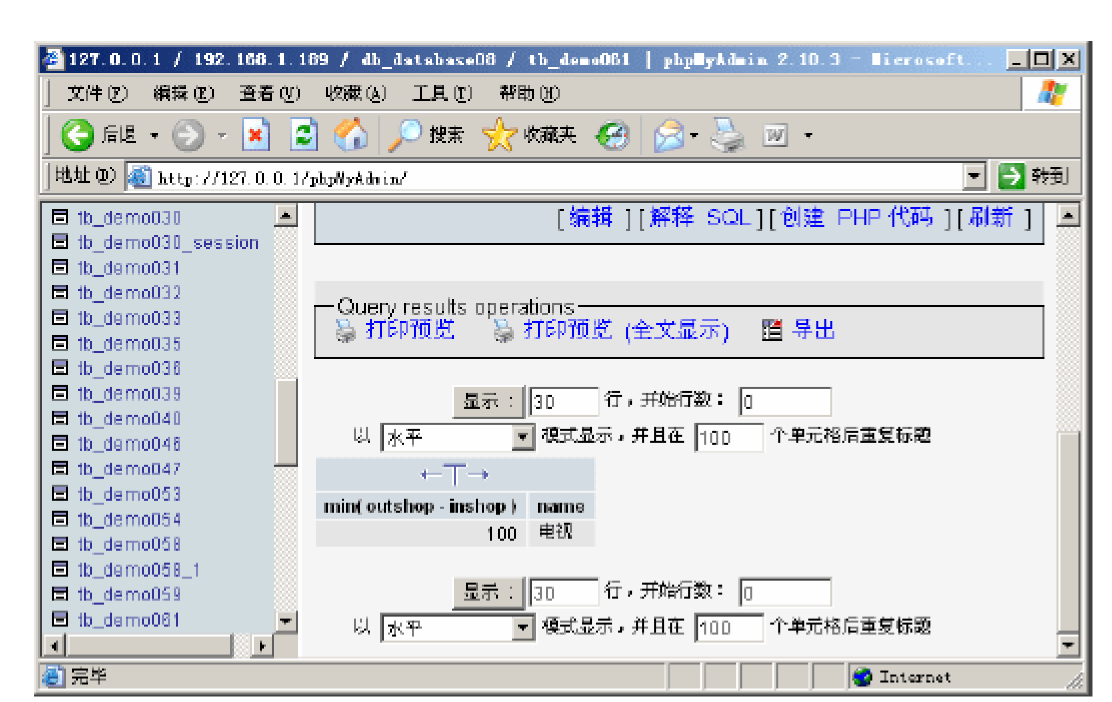
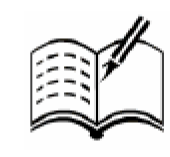


图 4.27　查询结果





问题 119　如何使用聚集函数 MAX 求销售利润最高的商品？

问题阐述

在商品销售管理系统中，商品的销售利润统计模块是必不可少的。那么，如何使用聚集函数 MAX 求销售利润最高的商品呢？

专家解答

在 SQL 语言中，应用 max()函数获取字段中的最大值，函数的语法如下：

```
max([all | distinct] expression)
```

参数说明：

❶ all：表示对指定字段的所有值进行聚集函数运算，all 为默认值，如果不加参数 all 或 distinct 则表示对指定字段的所有记录进行聚集运算。

❷ distinct：表示对指定字段的所有非重复记录进行求最大值。

❸ expression：是精确数字或近似数字数据类型分类（bit 数据类型除外）的表达式。

例如，查询 2005 年销售额最多的商品信息。

首先，进入 phpMyAdmin 中，选择 db_database08 数据库，单击“SQL”按钮，进入输入查询语句页面，如图 4.28 所示。

输入完查询语句后，单击“执行”按钮，即可显示查询结果，如图 4.29 所示。

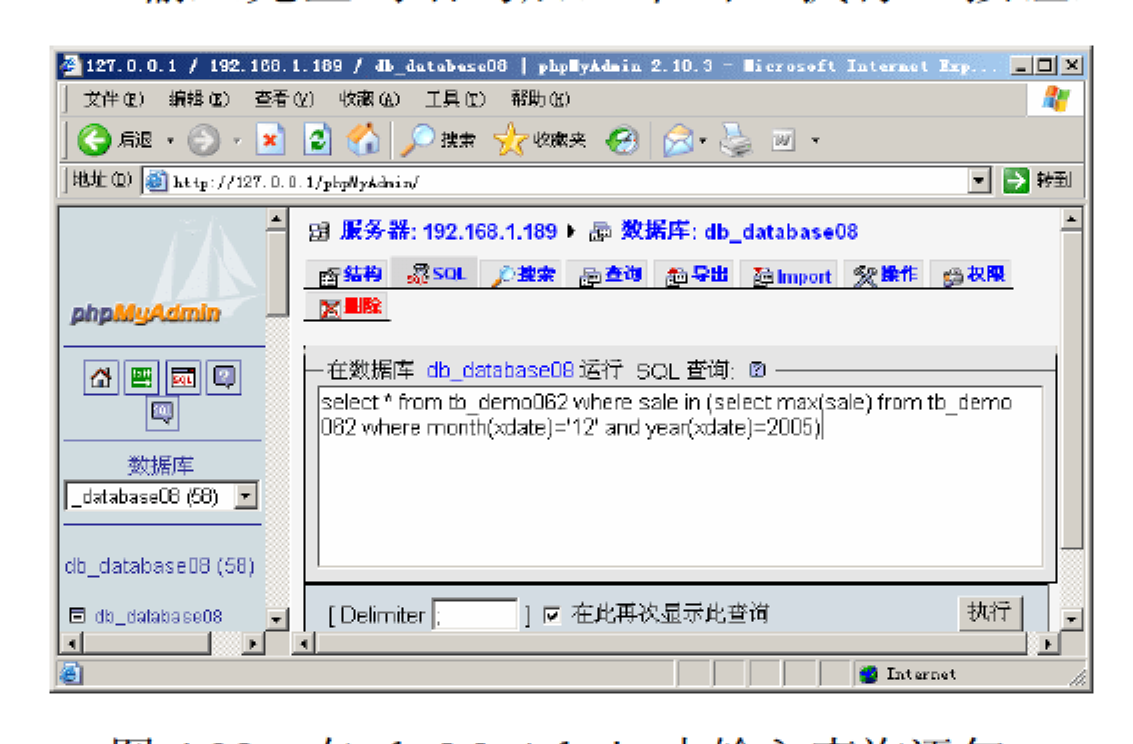


图 4.28　在 phpMyAdmin 中输入查询语句

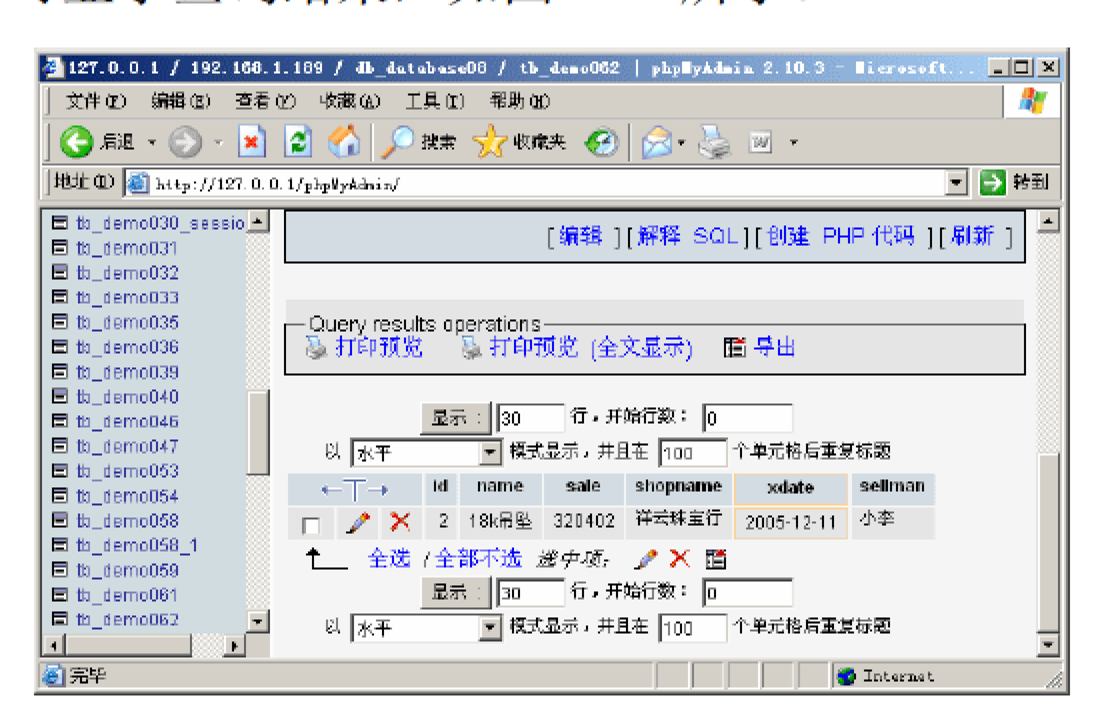


图 4.29　查询结果

问题 120　如何使用 INSERT(s1,x,len,s2)函数替换指定字符？

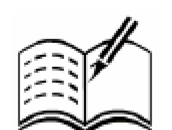
问题阐述

在 MySQL 数据库中有可能会遇到这样一种情况，在海量的数据记录中，存在一个错误的字符串，需要对这个错误的字符串进行修改，采用什么样的方法才是最快捷、最有效的呢？



专家解答

解决此问题，最理想的方法就是应用 MySQL 中的字符串函数 INSERT()，实现对指定字段中的字符串进行替换操作。





INSERT(s1,x,len,s2)函数可以将字符串 s1 中 x 位置开始长度为 len 的字符串替换为 s2 字符串。

例如，使用 INSERT 函数将 tb_book 表中 marker 字段中的第 3 个字符开始的两个字符替换为 book 字符。

首先，进入 phpMyAdmin 中，选择 db_database08 数据库，单击“SQL”按钮，进入输入查询语句页面，如图 4.30 所示。

输入完查询语句后，单击“执行”按钮，即可显示查询结果，如图 4.31 所示。

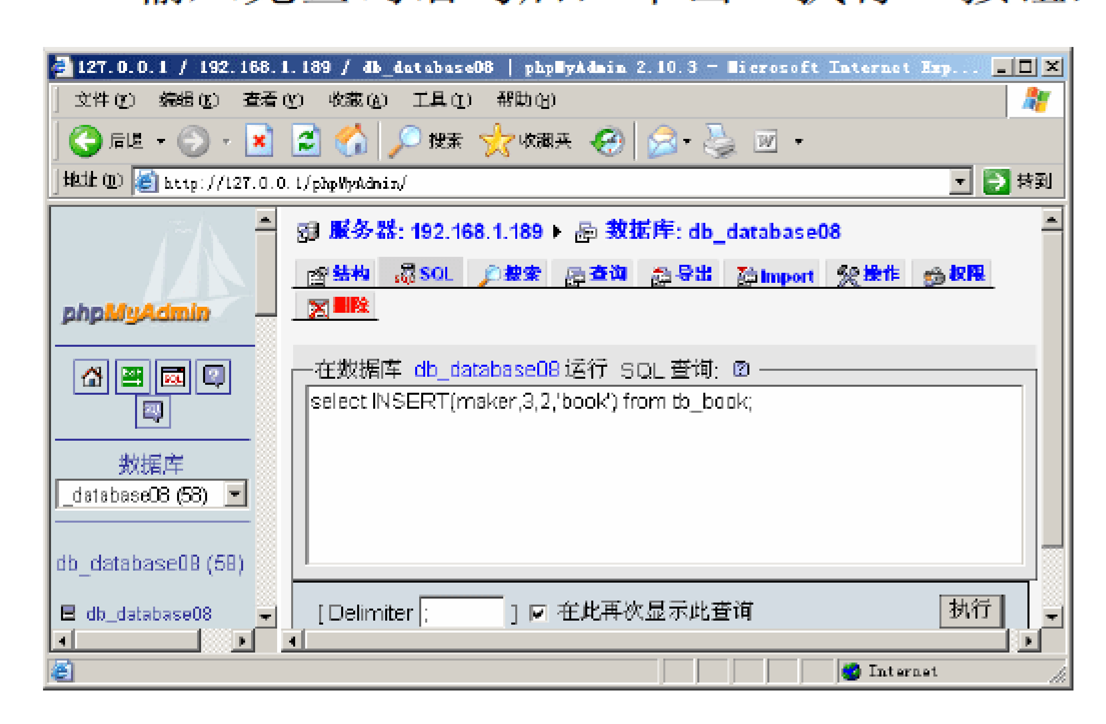


图 4.30　在 phpMyAdmin 中输入查询语句

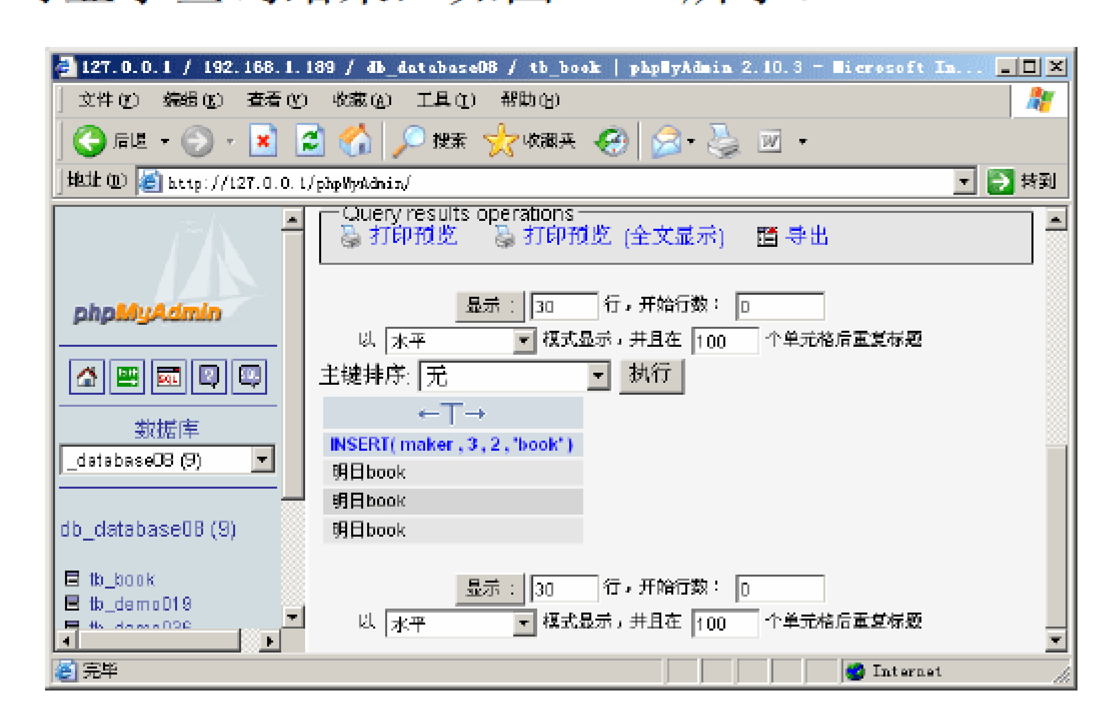


图 4.31　查询结果

专家点评

当数据库中的特定数据需要进行批量更新时，使用 INSERT()函数可以很好地完成操作。

问题 121　如何使用 UPPER(s)、UCASE(s)函数将字母变成大写？

问题阐述

如何使用 UPPER(s)、UCASE(s)函数将字母变成大写呢？

专家解答

通过 MySQL 中的 UPPER(s)函数和 UCASE(s)函数可以将字母变换成大写。

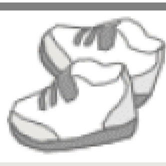
例如，下面将 tb_demo019 表中 name 字段中的英文全部变成大写。

首先，进入 phpMyAdmin 中，选择 db_database08 数据库，单击“SQL”按钮，进入输入查询语句页面，如图 4.32 所示。

输入完查询语句后，单击“执行”按钮，即可显示查询结果，如图 4.33 所示。

专家点评

通过 lower(s)和 lcase(s)函数可以将大写字母转换为小写。



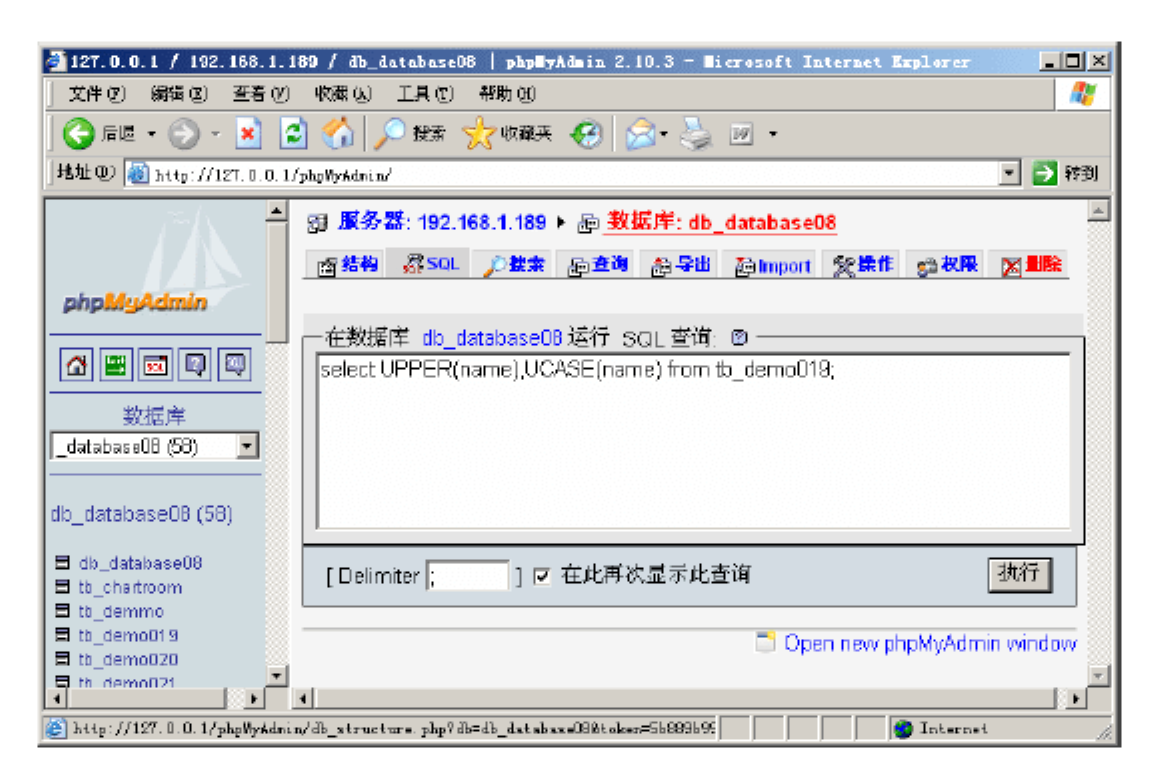


图 4.32　在 phpMyAdmin 中输入查询语句

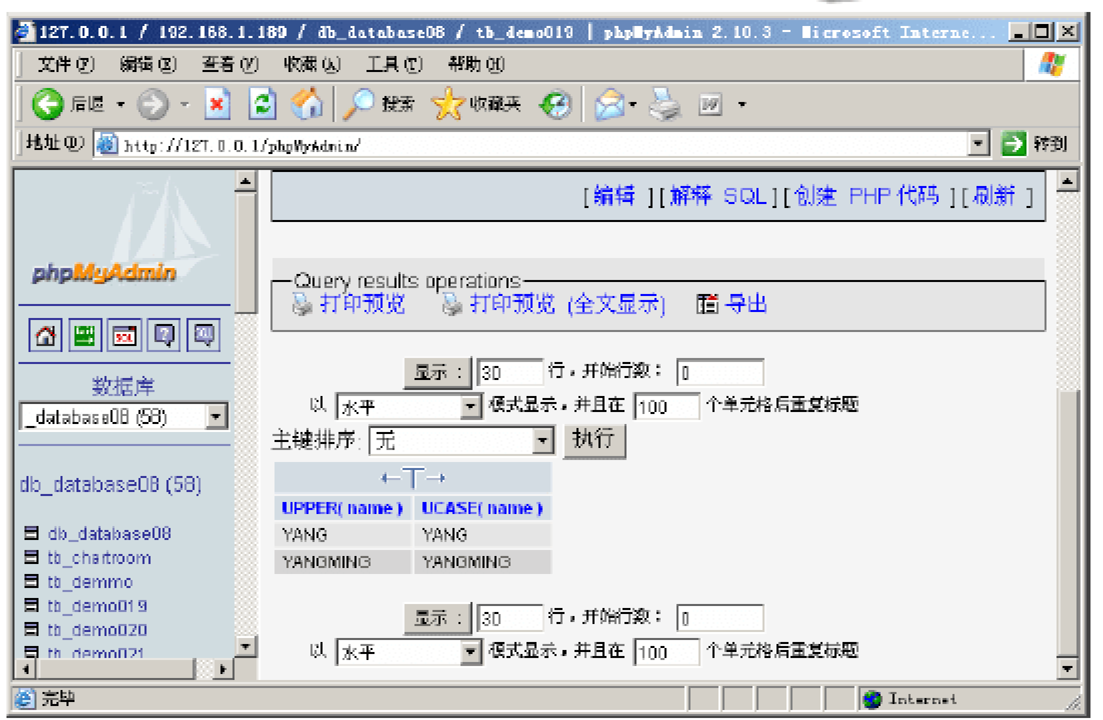


图 4.33　查询结果



Note

问题 122　如何利用 CSS 样式实现分页打印？

问题阐述

在制作数据打印程序时，对于多数据（指的是一页纸不能全部打印完毕的数据）通常采用分页打印。那么，如何利用 CSS 样式实现分页打印呢？

专家解答

利用 CSS 样式实现分页打印，其主要应用 thead 标记、tfoot 标记和 page-break-after 属性。

（1）thead 标记

thead 用于设置表格的表头。

（2）tfoot 标记

tfoot 用于设置表格的表尾。

（3）page-break-after 属性

page-break-after 属性在打印文档时发生作用，用于进行分页打印，但是对于
和<hr>对象不起作用。其语法格式如下：

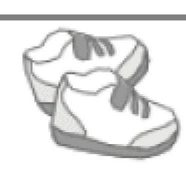
```
page-break-after： auto | always | avoid | left | right | null
```

参数说明：

❶ page-break：打印时在样式控制的对象前后换页。
❷ after：设置对象后出现页分隔符。设置为 always 时，始终在对象之后插入页分隔符。
❸ auto：需要在对象之后插入页分隔符时插入。
❹ always：始终在对象之后插入页分隔符。
❺ avoid：未支持。避免在对象后面插入分隔符。
❻ left：未支持。在对象后面插入页分隔符，直到它到达一个空白的左页边。
❼ right：未支持。在对象后面插入页分隔符，直到它到达一个空白的右页边。
❽ null：空白字符串。取消了分隔符设置。

应用示例

利用 CSS 样式实现分页打印。其具体步骤如下：



（1）编写用于控制指定内容不打印的 CSS 样式，代码如下：

```
@media print{
.bgnoprint{
    background:display:none;
}
.noprint{
    display:none
}
}
```

（2）应用 include 命令连接数据源文件，并应用 do…while 循环语句输出图书信息到浏览器，同时设置好表头、表尾及打印分页，关键代码如下：

```
<?php include "conn/conn.php"; ?>
<table width="99%" border="0" cellspacing="0" cellpadding="0">
    <tr>
      <td height="27" align="center" style=" font-size:14px;"><b>图书信息查询</b></td>
   </tr>
 </table>
   <table width="98%" border="0" cellpadding="0" cellspacing="1" bgcolor="#000000" bordercolor=
"#FFFFFF" bordercolordark="#000000" bordercolorlight="#FFFFFF" >
    <thead style="display:table-header-group;"> <!--设置表头-->
    <tr bgcolor="#EFEFEF">
        <td width="6%" height="20" align="center">编号</td>
        <td width="27%" align="center">图书名称</td>
        <td width="23%" align="center">内容简介</td>
        <td width="8%" align="center">定价</td>
        <td width="10%" align="center">作者</td>
        <td width="15%" align="center">出版社</td>
        <td width="11%" align="center">发行时间</td>
    </tr>
      </thead>
    <!--控制分页-->
  <?php
$sql=mysql_query("select * from tb_book");
$info=mysql_fetch_array($sql);
$row=1;
    do{
    ?>
    <tr align="center" <?php if($row==2){ ?>style="page-break-after:always"<?php } ?>>
        <td bgcolor="#FFFFFF"><?php echo $info[id];?></td>
        <td height="25" align="left" bgcolor="#FFFFFF"> <?php echo $info[bookname];?></td>
        <td align="left" bgcolor="#FFFFFF"> <?php echo $info[synopsis];?></td>
        <td bgcolor="#FFFFFF"><?php echo $info[price];?></td>
        <td bgcolor="#FFFFFF"><?php echo $info[maker];?></td>
        <td bgcolor="#FFFFFF"><?php echo $info[publisher];?></td>
        <td bgcolor="#FFFFFF"><?php echo $info[issuDate];?></td>
     </tr>
<?php
$row++;
    }while($info=mysql_fetch_array($sql))
?>
<!--设置表尾-->
```



```
<tfoot style="display:table-footer-group; border:none;"><tr><td></td></tr></tfoot>
</table>
```

（3）建立 HTML 的 object 标签，调用 WebBrowser 控件，代码如下：

```
<object id="Wb" classid="ClSID:8856F961-340A-11D0-A96B-00C04Fd705A2" width="0" height="0">
</object>
```

（4）建立相关的打印超链接，并调用 WebBrowser 控件的相应参数实现打印预览及打印功能，代码如下：

```
<table width="99%" height="25" border="0" cellpadding="0" cellspacing="0">
    <tr>
      <td height="25" align="right" class="noprint">
<a href="#" onClick="document.all.Wb.Execwb(7,1)">打印预览</a>
<a href="#" onClick="document.all.Wb.Execwb(6,1)">打印</a>
</td>
  </tr>
</table>
```

运行本示例，如图 4.34 所示，单击“打印预览”超链接，可以查看打印结果，如图 4.35 所示；单击“打印”超链接即可进行分页打印。



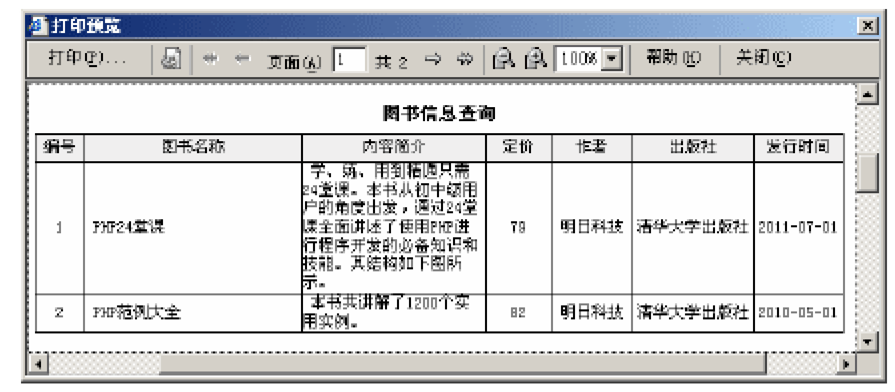


图 4.34　利用 CSS 样式分页打印

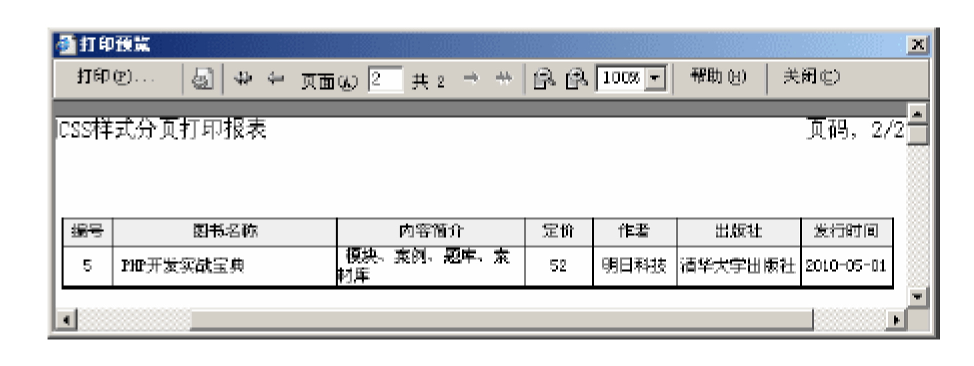
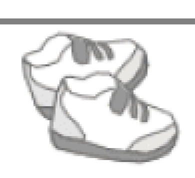


图 4.35　打印预览效果

问题 123　如何使用 RTRIM(s)函数去掉字符串结尾处的空格?

问题阐述

如何使用 RTRIM(s)函数去掉字符串结尾处的空格呢？



专家解答

RTRIM(s)函数主要用于去掉字符串结尾处的空格。

例如，下面将 tb_demo019 表中 name 字段值结尾处的空格去掉。

首先，进入 phpMyAdmin 中，选择 db_database08 数据库，单击“SQL”按钮，进入输入查询语句页面，如图 4.36 所示。

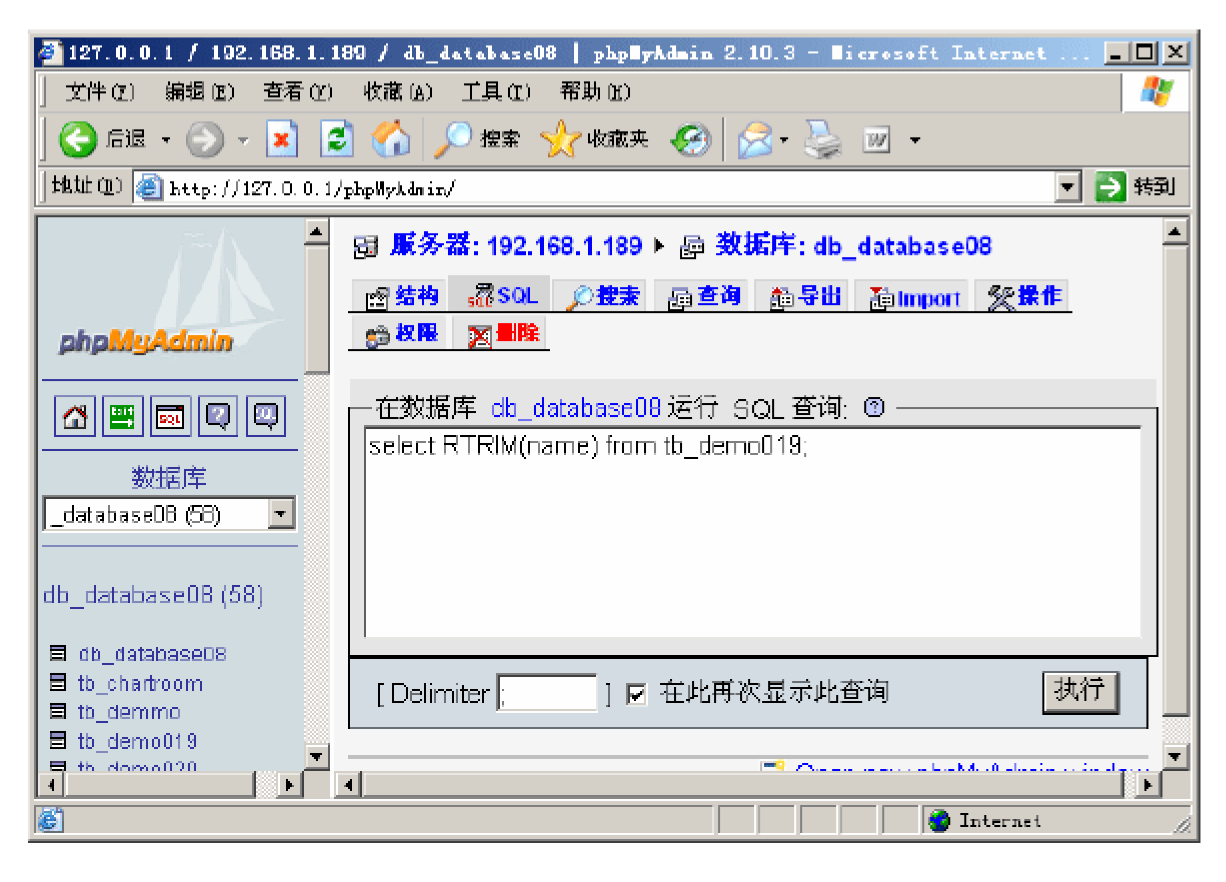


图 4.36 在 phpMyAdmin 中输入查询语句

输入完查询语句后，单击“执行”按钮，即可显示查询结果，去除尾空格前后如图 4.37 和图 4.38 所示。

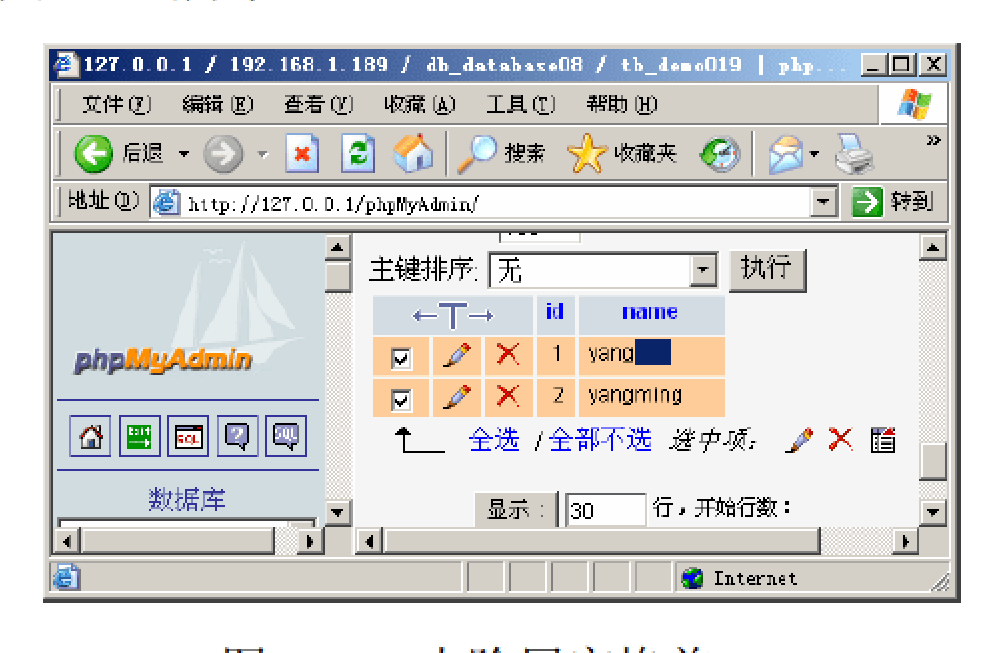


图 4.37 去除尾空格前

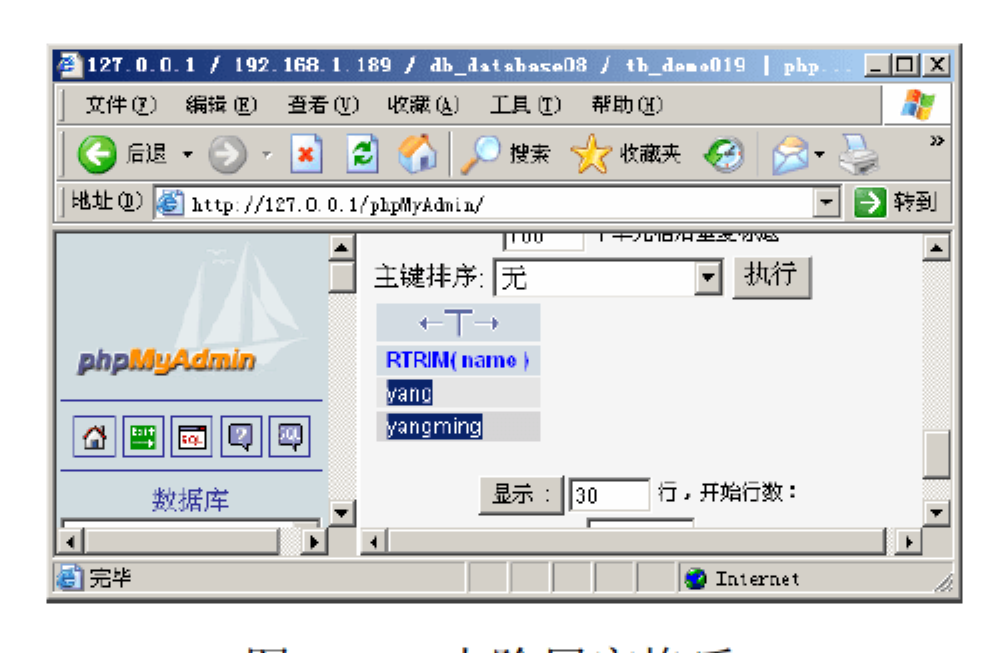


图 4.38 去除尾空格后

问题 124 MySQL 数据库的备份和恢复

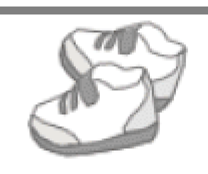
问题阐述

数据库越大，数据备份的问题就越让人头疼。这确实像是噩梦一般：如果服务器崩溃，所有东西都丢失了，后果是不堪设想的。那么我们必须掌握最有效的数据库备份和恢复的方法，在灾难发生时可以及时进行补救，那么什么样的方法才是最有效的数据库备份和恢复的方法呢？

专家解答

1. 通过 mysqldump 命令备份数据库

在命令模式下完成对数据的备份，使用的是 MYSQLDUMP 命令。通过该命令可以将数据





以文本文件的形式存储到指定的文件夹下。

通过 MYSQLDUMP 命令备份整个数据库的操作步骤如下：

（1）选择“开始”菜单中的“运行”命令，如图 4.39 所示。

（2）在如图 4.40 所示的对话框中输入“cmd”，单击“确定”按钮，进入命令模式。



图 4.39　选择运行命令

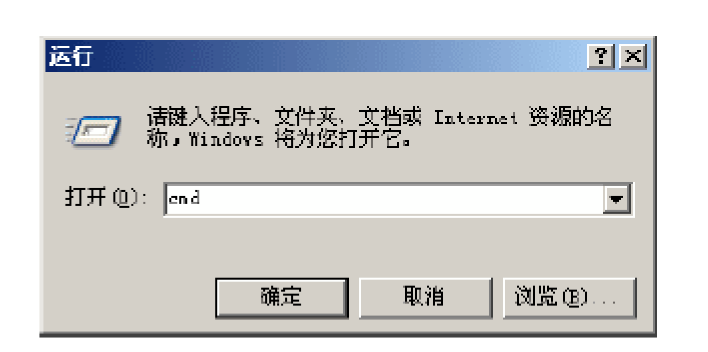


图 4.40　进入命令模式

（3）在命令模式中直接输入“mysqldump -uroot -proot db_database16>F:\db_database16.txt”，然后按回车键即可，如图 4.41 所示。

其中，-uroot 中的“root”是用户名，-proot 中的“root”是密码，“db_database16”是数据库名，“F:\db_database16.txt”是数据库备份存储的位置。

说明：

在输入命令的过程中，在“-uroot”中是没有空格的，在该命令的结尾处也没有任何的结束符，只要按回车键即可。

2. 通过 mysql 命令恢复数据库

既然可以对数据库进行备份，那么就一定可以对数据库文件进行恢复操作。执行数据库的恢复操作使用的是 MySQL 命令。其命令格式如下：

```
mysql -uroot -proot db_database <F:\db_database16.txt
```

其中 MySQL 是使用的命令，-u 后的 root 代表用户名，-p 后的 root 代表密码，db_database 代表数据库名（或表名），“<”号后面的“F:\db_database16.txt”是数据库备份文件存储的位置。

下面介绍实现数据库恢复的操作步骤：

（1）选择“开始”菜单中的“运行”命令。

（2）在弹出的对话框中输入“cmd”，单击“确定”按钮，进入命令模式。

（3）在命令模式中直接输入“mysql -uroot -proot db_database16 <F:\db_database16.txt”，然后按回车键即可，如图 4.42 所示。

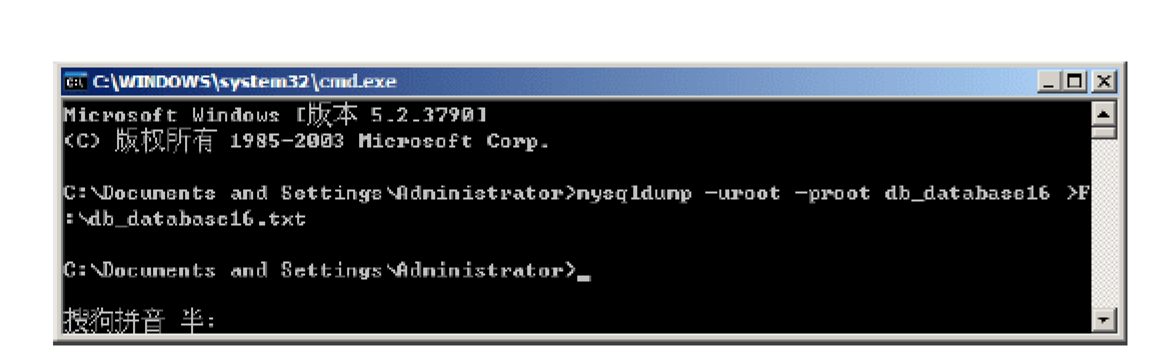


图 4.41　通过命令备份 db_database16 数据库

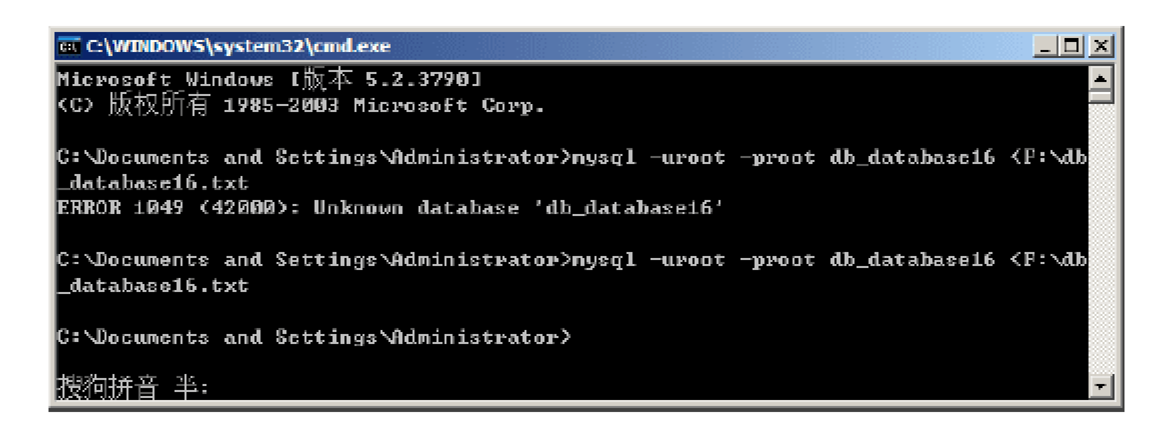


图 4.42　通过命令恢复 db_database16 数据库

其中，-uroot 中的“root”是用户名，而-proot 中的“root”是密码，“db_database16”是要恢复的数据库名，“F:\db_database16.txt”是数据库备份文件存储的位置。

说明：

在进行数据库的恢复时，在用户的 MySQL 数据库中必须存在一个空的、将要恢复的数据



库。在 MySQL 服务器中创建一个 db_database16 数据库，然后，再执行数据的恢复操作。



Note

专家点评

无论作为数据库管理人员、维护人员还是开发人员，掌握如何备份数据库都是十分必要的。数据库备份工作既可以提高系统的安全性，又给管理者在维护工作上带来了极大方便。所以，如何备份 MySQL 数据库也在 MySQL 数据库管理中占有一席之地。

问题 125　如何在 MySQL 中创建存储过程？

问题阐述

自 MySQL 5.0 开始，MySQL 就支持存储过程。存储过程是一些被用户定义的 SQL 语句集合。一个存储程序是可以被存储在服务器中的一套 SQL 语句。存储过程可以被程序、触发器或另一个存储过程调用。其具有执行速度快、提高系统性能、确保数据库安全等诸多优点。那么如何在 MySQL 中创建存储过程呢？

专家解答

1. 存储过程创建解析

在 MySQL 中，创建存储过程的基本形式如下：

```
CREATE PROCEDURE sp_name ([proc_parameter[,…]])
    [characteristic…] routine_body
```

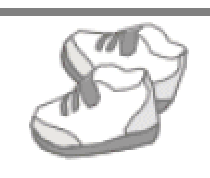
其中 sp_name 参数是存储过程的名称；proc_parameter 表示存储过程的参数列表；characteristic 参数指定存储过程的特性；routine_body 参数是 SQL 代码的内容。

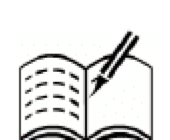
proc_parameter 中的参数由 3 部分组成，它们分别是输入输出类型、参数名称和参数类型。其形式为[IN|OUT|INOUT]param_name type。其中 IN 表示输入参数；OUT 表示输出参数；INOUT 表示既可以输入也可以输出；param_name 参数是存储过程参数名称；type 参数指定存储过程的参数类型，该类型可以为 MySQL 数据库的任意数据类型。

一个存储过程包括名字，参数列表，还可以包括很多 SQL 语句集。下面创建一个存储过程，其代码如下：

```
delimiter //
create procedure proc_name (in parameter integer)
begin
declare variable varchar(20);
if parameter=1 then
set variable='MySQL';
else
set variable=’PHP’;
end if;
insert into tb (name) values (variable);
end;
```

MySQL 中存储过程的建立以关键字 create procedure 开始，后面仅跟存储过程的名称和参数。MySQL 的存储过程名称不区分大小写，如 PROCE1()和 proce1()代表同一存储过程名。存





储过程名或存储函数名不能与 MySQL 数据库中的内建函数重名。

MySQL 存储过程的语句块以 begin 开始，以 end 结束。语句体中可以包含变量的声明、控制语句、SQL 查询语句等。由于存储过程内部语句要以分号结束，所以在定义存储过程前，应将语句结束标志“;”更改为其他字符，并且应降低该字符在存储过程中出现的机率，更改结束标志可以用关键字“delimiter”定义，例如：

```
mysql>delimiter //
```

存储过程创建之后，可用如下语句进行删除，参数 proc_name 指存储过程名。

```
drop procedure proc_name
```

2. 应用示例

下面创建一个名称为 count_of_sutdent 的存储过程。首先，创建一个名称为 students 的 MySQL 数据库，然后创建一个名为 studentinfo 的数据表。数据表结构如表 4.3 所示。

表 4.3　studentinfo 数据表结构

字　段　名	类型（长度）	默　　认	额　　外	说　　明
sid	INT(11)		auto_increment	主键自增型sid
name	VARCHAR(50)			学生姓名
age	VARCHAR(11)			学生年龄
sex	VARCHAR(2)	M		学生性别
tel	BIGINT(11)			联系电话

创建一个名称为 count_of_student 的存储过程，统计 studentinfo 数据表中的记录数。代码如下：

```
delimiter //
create procedure count_of_student(OUT count_num INT)
reads sql data
begin
select count(*) into count_num from studentinfo;
end
//
```

在上述代码中，定义一个输出变量 count_num。存储过程应用 SELECT 语句从 studentinfo 表中获取记录总数。最后将结果传递给变量 count_num。存储过程的执行结果如图 4.43 所示。

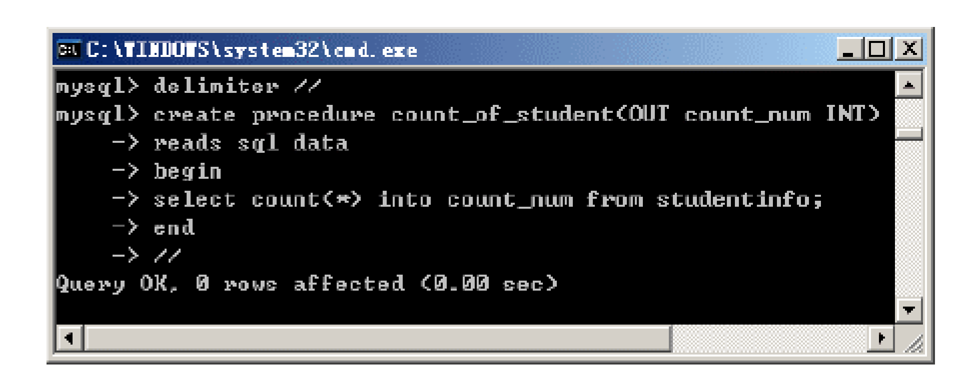


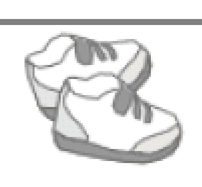
图 4.43　创建存储过程 count_of_student

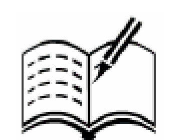
专家点评

MySQL 中使用 CALL 语句来调用存储过程。调用存储过程后，数据库系统将执行存储过程中的语句。然后将结果返回给输出值。CALL 语句的基本语法形式如下：

```
CALL sp_name([parameter[,…]]);
```

其中 sp_name 是存储过程的名称，parameter 是存储过程的参数。





Note

问题 126　如何将 Access 数据导入 SQL Server 数据库中？

问题阐述

如何将 Access 数据库中的数据导入 SQL Server 中呢？

专家解答

本例将 Access 数据库文件 db_medicine.mdb 导入 SQL Server 数据库文件 db_database（该数据库为空，没有创建任何数据表）中，步骤如下：

（1）打开 SQL Server 数据库，右键选择接收 Access 数据库文件的数据库（此处选择 db_database 数据库），在弹出的菜单中选择"任务"→"导入数据"命令，如图 4.44 所示。

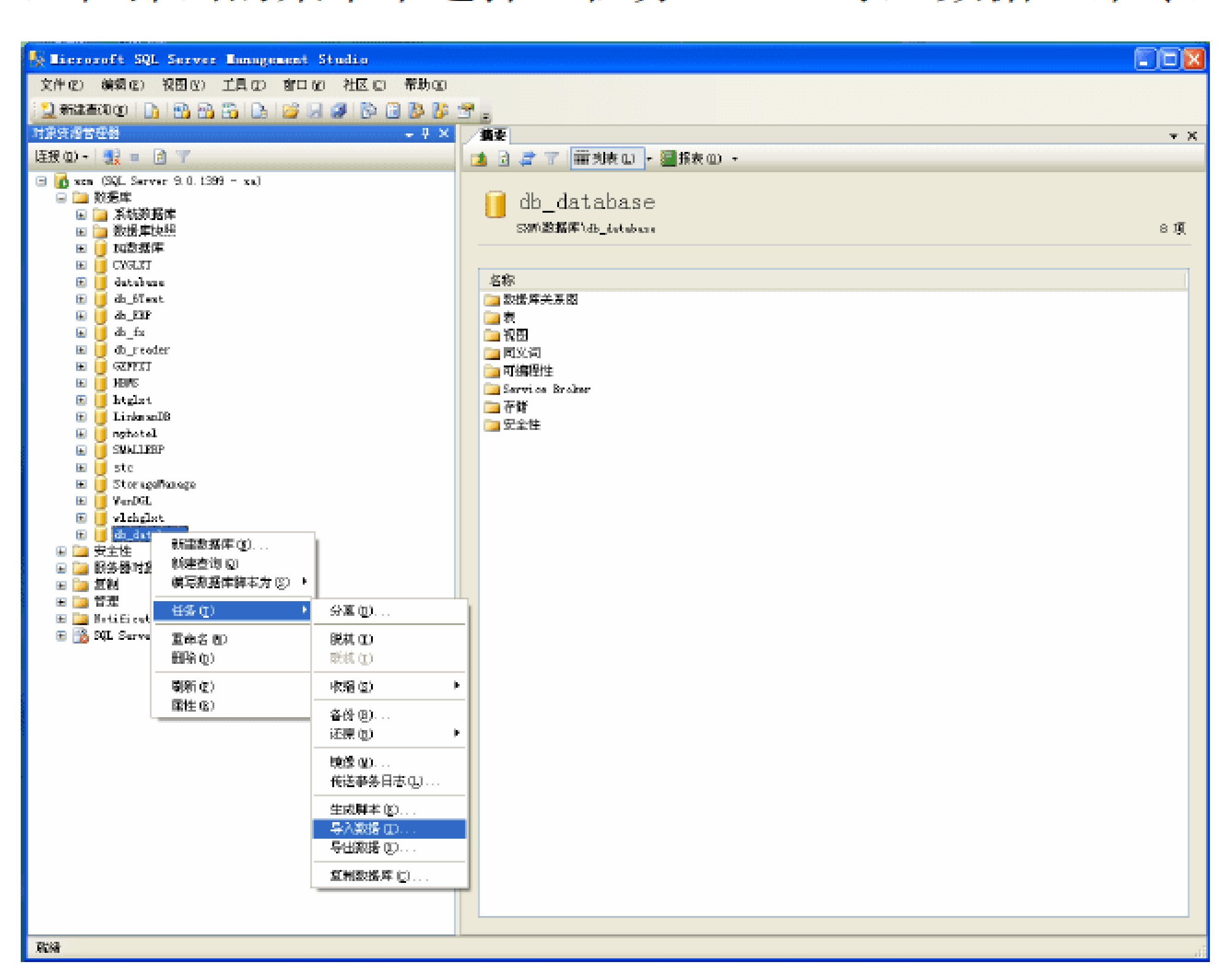


图 4.44　选择导入数据菜单

（2）打开"SQL Server 导入和导出向导"窗口，单击"下一步"按钮，如图 4.45 所示。

（3）选择导入数据库的数据源（此处为 Microsoft Access），并选择要导入的数据库文件，如图 4.46 所示。



图 4.45　"SQL Server 导入和导出向导"窗口

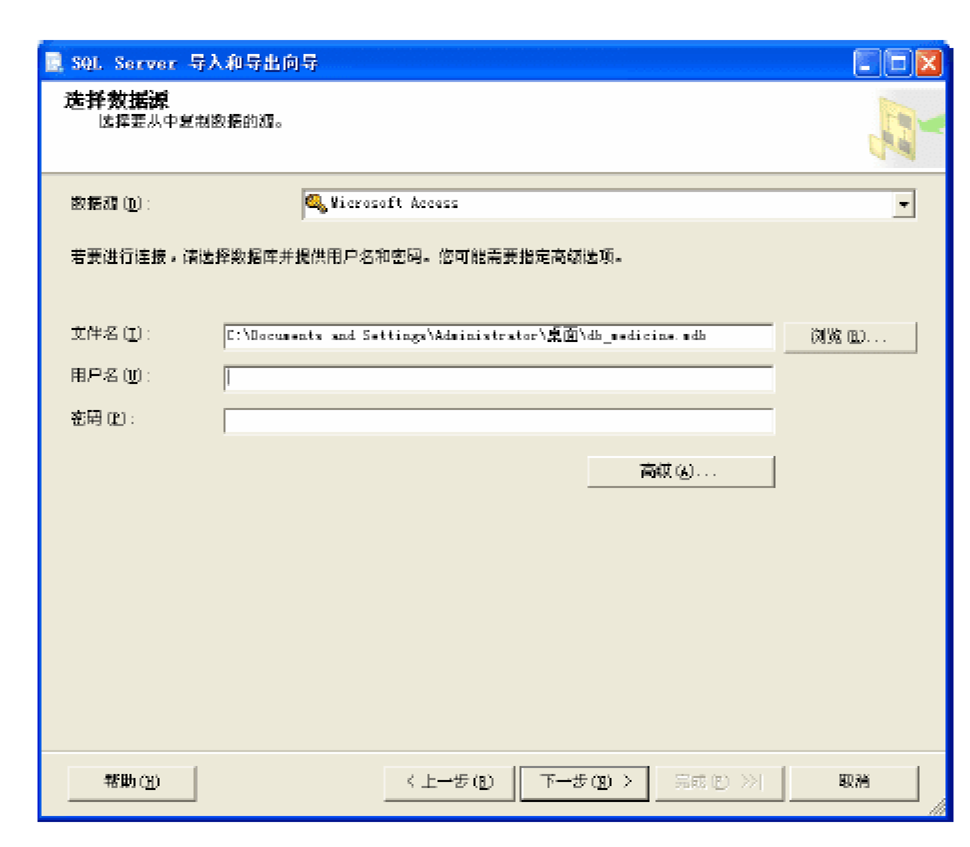
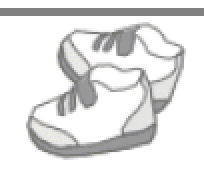
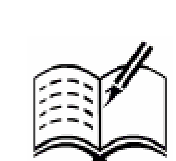


图 4.46　选择要导入的数据库





（4）选择目标数据库文件，如图 4.47 所示。

（5）指定表复制或查询，此处选中“复制一个或多个表或视图的数据”单选按钮，如图 4.48 所示。

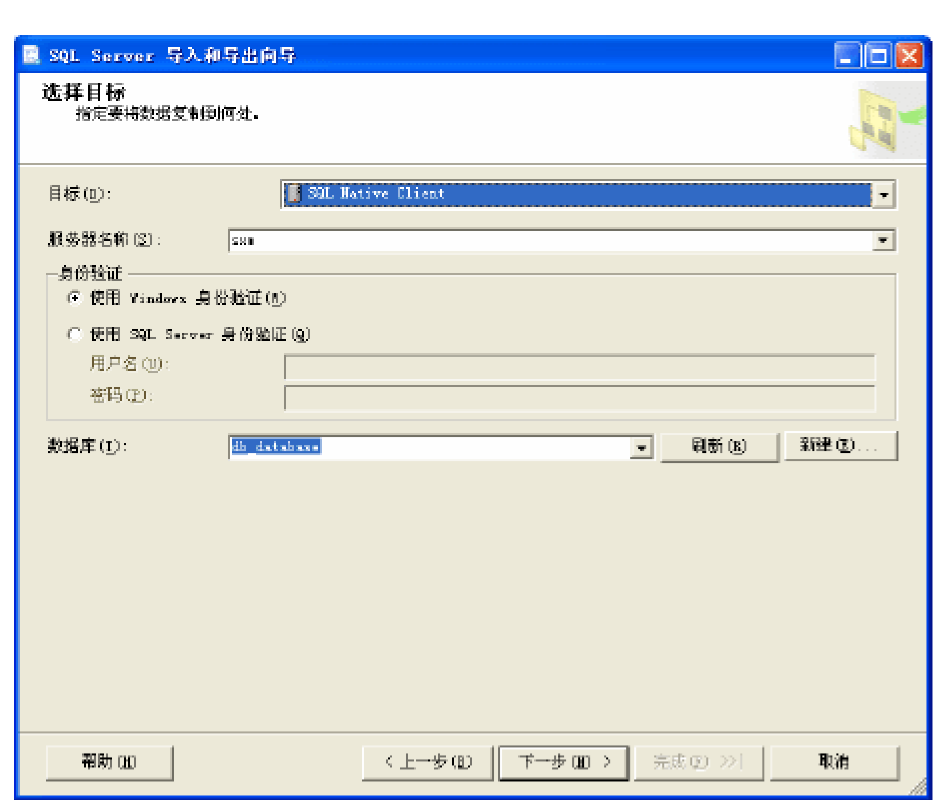


图 4.47　选择目标数据库文件

图 4.48　指定表复制或查询选项

（6）选择要复制的数据表（此处选择“全选”），单击“下一步”按钮，如图 4.49 所示。

（7）保存并立即执行，单击“下一步”按钮，如图 4.50 所示。

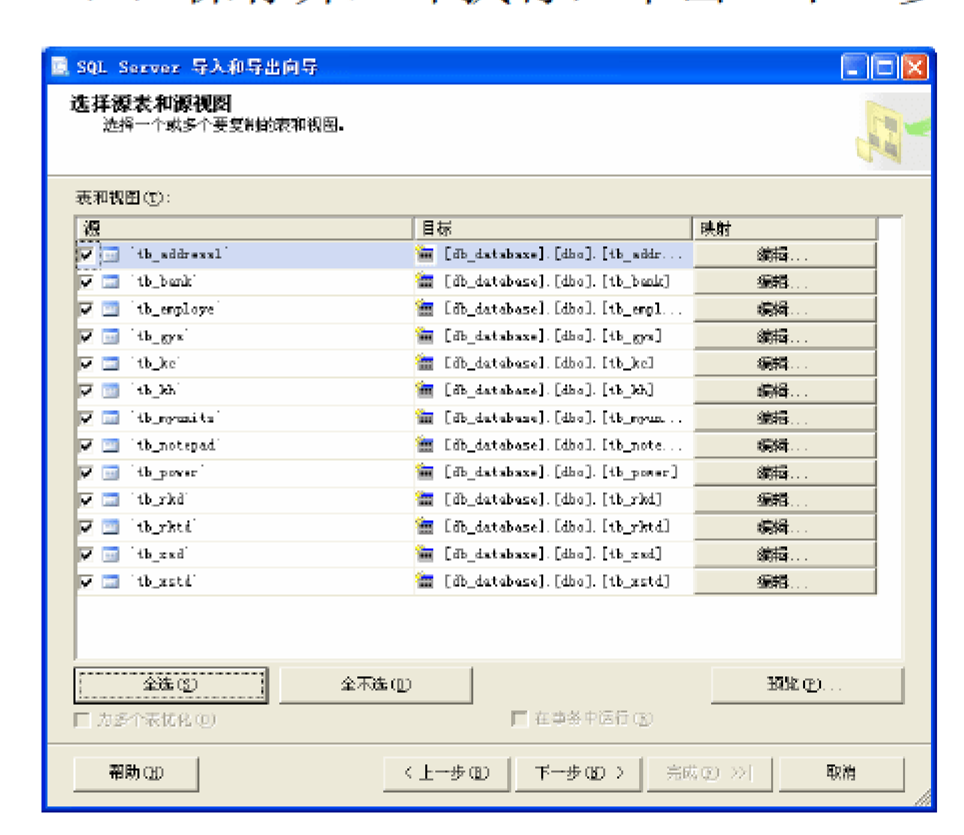


图 4.49　选择要复制的数据表

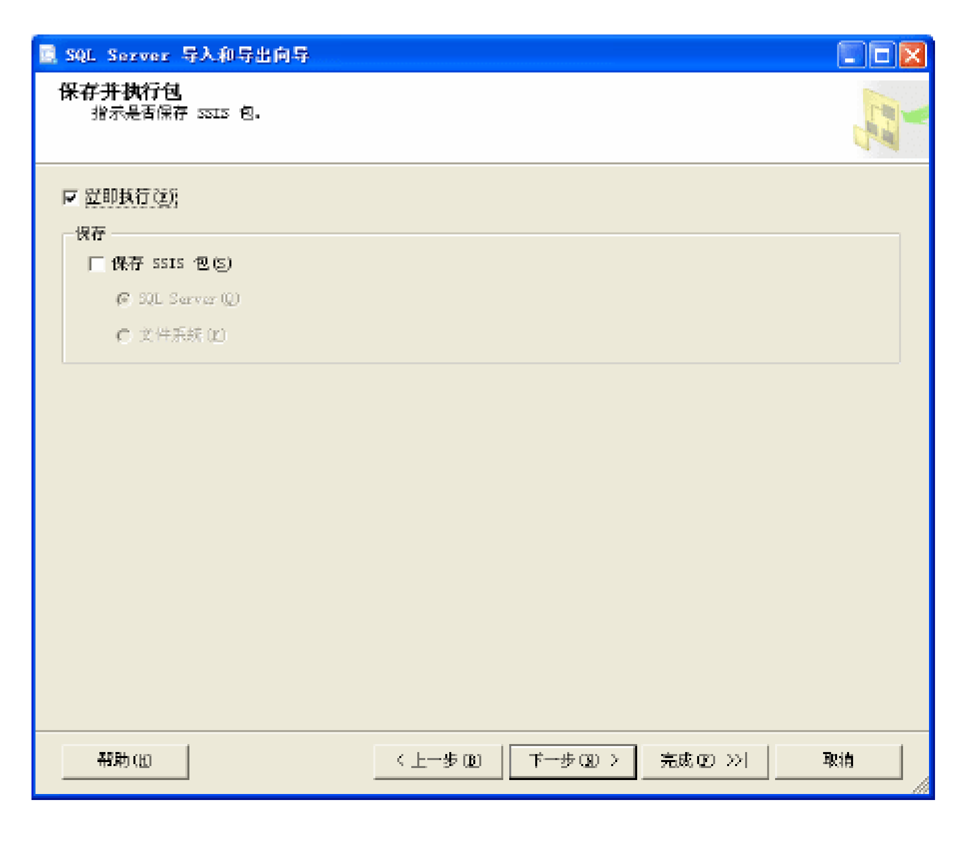


图 4.50　保存并立即执行

（8）验证向导中选择的选项，单击“完成”按钮，如图 4.51 所示。

（9）正在执行复制操作，如图 4.52 所示。

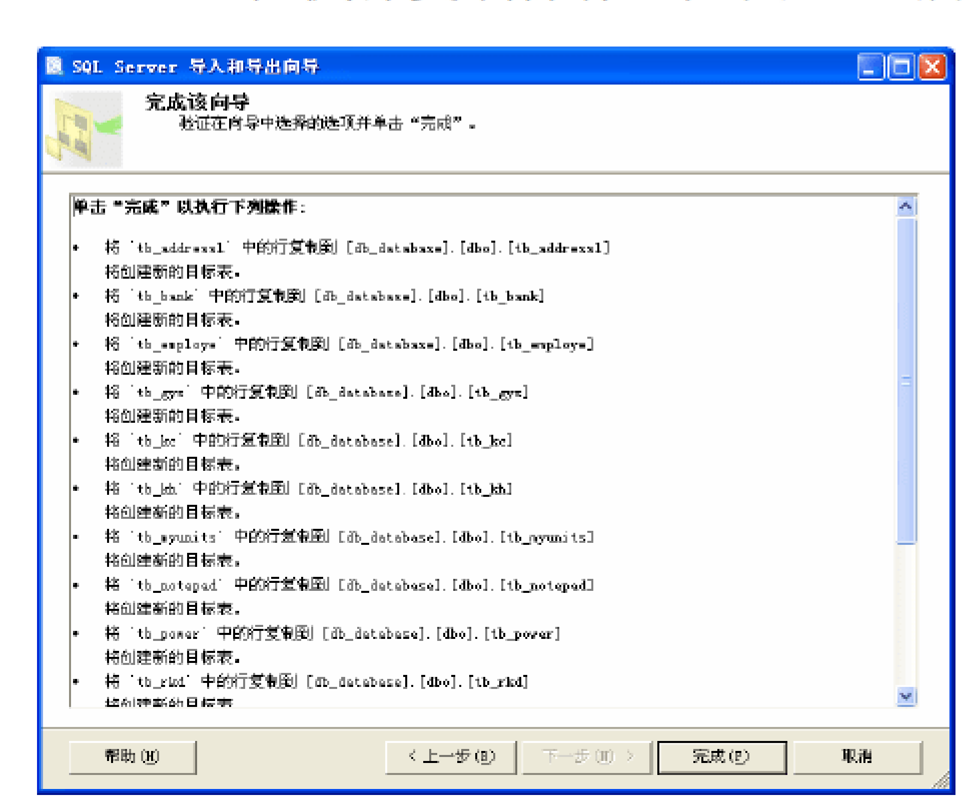


图 4.51　验证向导中选择的选项

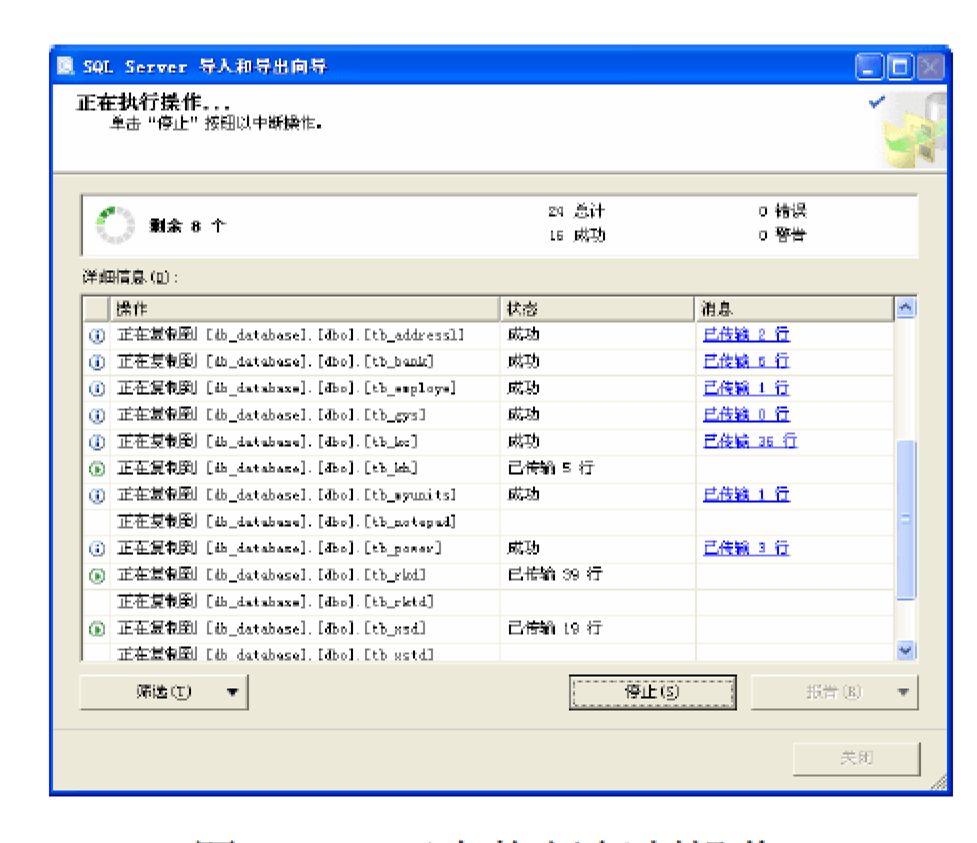
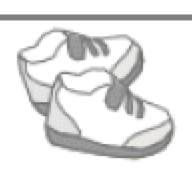
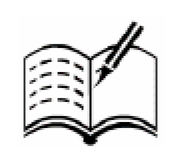


图 4.52　正在执行复制操作







（10）复制完成后，单击“关闭”按钮，如图 4.53 所示。

复制后的 db_database 数据库中的数据表文件如图 4.54 所示。

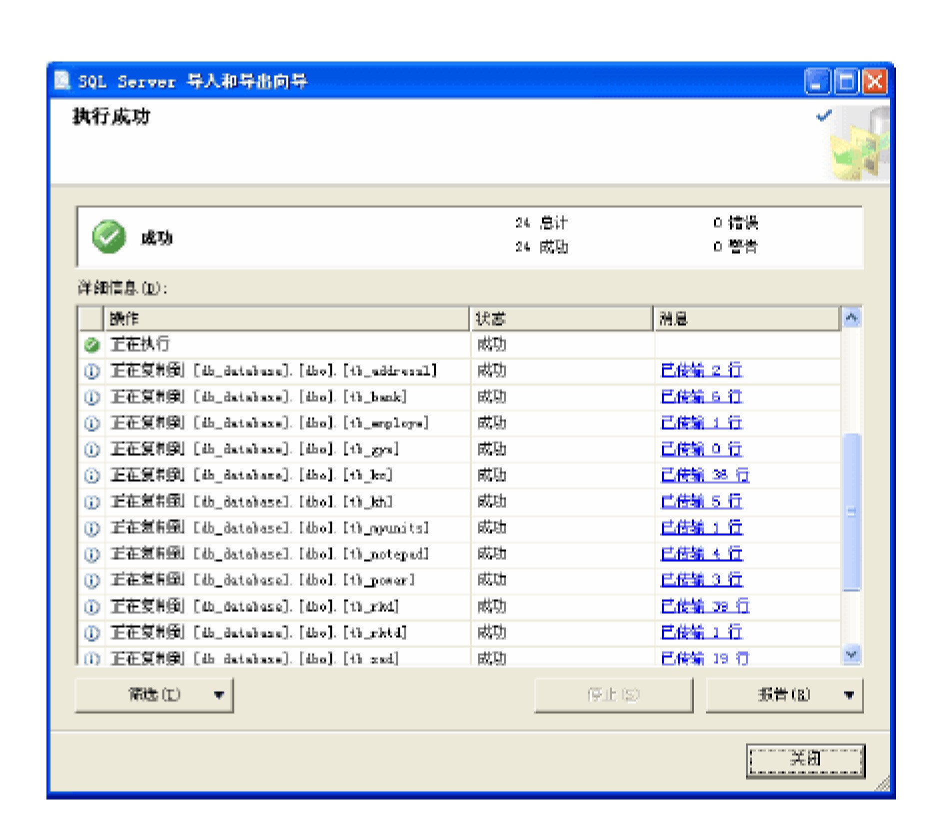


图 4.53　数据表复制成功

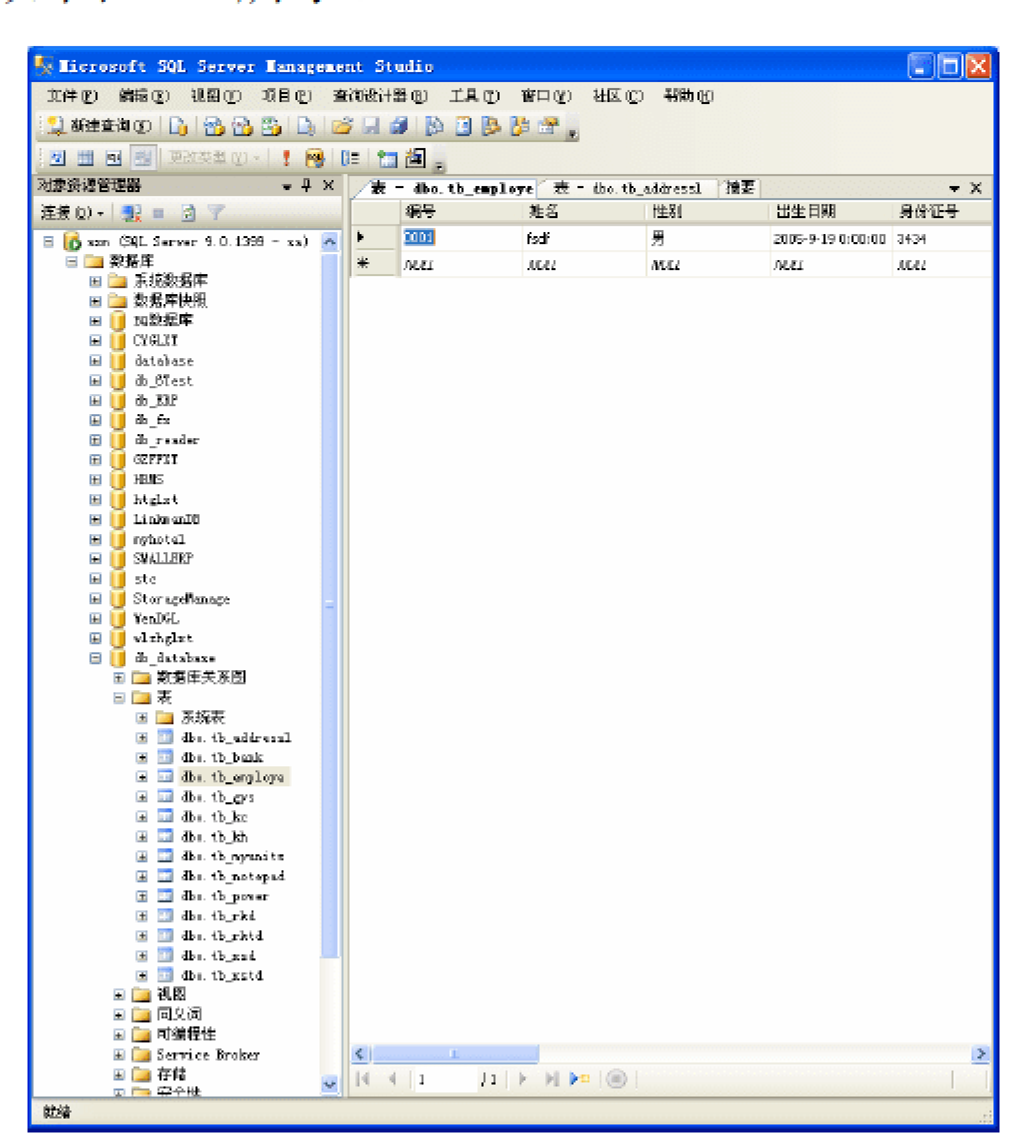


图 4.54　复制后的 SQL Server 数据库文件

问题 127　如何解决 PHP 不支持 SQL Server 的问题？

问题阐述

PHP 有时会与 SQL Server 数据库合作开发，初次合作需要配置一下，否则 PHP 会不支持 SQL Server 服务器。那么如何解决 PHP 不支持 SQL Server 的问题呢？

专家解答

（1）在 php.ini 文件中，将“;extension=php_mssql.dll”前的分号去掉，保存后重新启动 Apache 服务器。

（2）将 PHP 中提供的 ntwdblib.dll 文件复制到本机系统的 windows\system32 文件夹下，重新启动 Apache 服务器即可。

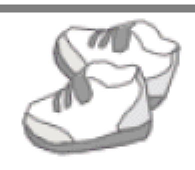
专家点评

ntwdblib.dll 文件是用于与 SQL Server 客户端的连接。如果想要 PDO 支持 SQL Server 数据库，那么在 php.ini 文件中还必须将“;extension=php_pdo.dll”和“;extension=php_pdo_mssql.dll”前的分号去掉。

问题 128　如何连接带密码的 Access 数据库？

问题阐述

虽然 Access 作为小型项目的开发首选，但是，商家不会因为项目小而不要求它的安全性。





所以给 Access 数据库设置密码是十分必要的。那么，在 PHP 中如何连接带密码的 Access 数据库呢？

专家解答

1. 设置 Access 数据库的密码

（1）设置 Access 数据库密码与操作其他数据库不同，必须以独占方式打开数据库。

（2）选择“工具”→“安全”→“设置数据库密码”命令，设置密码和确认密码，如图 4.55 所示。



图 4.55 设置并确认密码

2. 定义数据库的连接方法

```
<?php
    $conn = new com("adodb.connection");
    $connstr="driver={microsoft access driver (*.mdb)};Pwd=111; dbq=". realpath("data/db_database11.mdb");
    $conn->open($connstr);
?>
```

专家点评

当用户对 Access 数据库进行维护时，为了方便起见，可以暂时撤销 Access 数据库密码，方法是：打开数据库，输入密码并进入数据库，然后选择“工具”→“安全”→“撤销数据库密码”命令，根据提示再次输入密码，单击“确定”按钮即可撤销数据库密码。

问题 129 MySQL 和 Server 2000 数据库中数据类型的转换

问题阐述

在某些特殊的时候，可能要将 MySQL 数据库转换为 SQL Server 2000 数据库，这时问题也就随之而来。例如，在 MySQL 数据库中，创建的字段使用 varchar 类型，该类型的数据可以进行聚合运算，其返回的值是 string 类型的数据。但是，在 SQL Server 数据库中，创建的字段如果是 varchar 类型，则不能够进行聚合运算。另外，在 SQL Server 数据库中，int 型的数据将不能应用到 for 循环语句中，这些都是由于数据库的转换而带来的问题。

专家解答

针对 MySQL 和 Server 2000 数据库转换中的问题，其基本的策略就是实现对数据类型的转



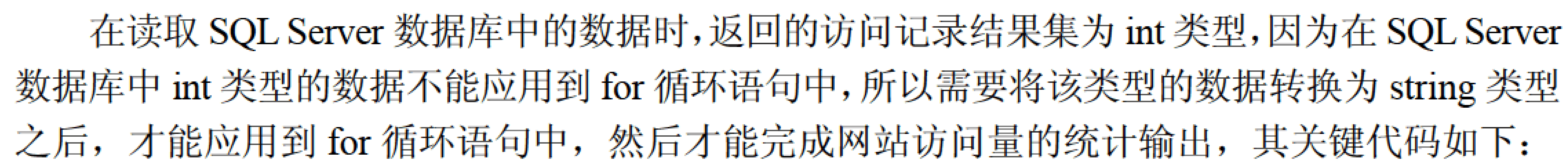
换，经过数据类型的转换就可以完成相应的操作了。

例如，在 SQL Server 数据库中创建的 varchar 类型数据，不能够进行聚合运算，但可以将其转换为 int 型进行计算。由于 int 型的数据不能应用到 for 循环语句中，所以必须将 int 型的数据转换为 string 类型，才能应用到 for 循环语句中。



Note

应用示例



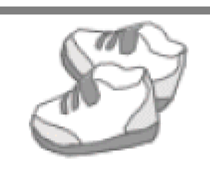
在读取 SQL Server 数据库中的数据时，返回的访问记录结果集为 int 类型，因为在 SQL Server 数据库中 int 类型的数据不能应用到 for 循环语句中，所以需要将该类型的数据转换为 string 类型之后，才能应用到 for 循环语句中，然后才能完成网站访问量的统计输出，其关键代码如下：

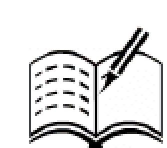
```
<?php
//以图形的形式输出数据库中的记录数
$query="select sum(counts) as ll from tb_count10 ";        //查询数据库中总的访问量
$results=mssql_query($query,$conn);
$fwl=mssql_result($results,0,'ll');
    echo "----------";
//对补位数字 0 的处理
$len=strlen($fwl);                                         //获取字符串的长度
$str=str_repeat("0",6-$len);                               //获取 6-$len 个数字 0
for($a=0;$a<strlen($str);$a++){                            //获取变量$str 的字符串长度
        $results=$str[$a];
        $results='<img src=images/0.gif>';
        echo $results;                                     //循环输出$result 的结果
    }
    //对数据库中数据的处理
settype($fwl,"string");                      //进行类型转换,将 INT 类型的数据转换为 STRING 类型
    for($i=0;$i<strlen($fwl);$i++){          //获取字符串的长度
        $result=$fwl[$i];
        switch($result){
            //如果值为"0",则输出 0.gif 图片
            case "0"; $ret[$i]="0.gif";break;
            case "1"; $ret[$i]="1.gif";break;
            case "2"; $ret[$i]="2.gif";break;
            case "3"; $ret[$i]="3.gif";break;
            case "4"; $ret[$i]="4.gif";break;
            case "5"; $ret[$i]="5.gif";break;
            case "6"; $ret[$i]="6.gif";break;
            case "7"; $ret[$i]="7.gif";break;
            case "8"; $ret[$i]="8.gif";break;
            case "9"; $ret[$i]="9.gif";break;
        }
        echo "<img src=images/".$ret[$i].".>";
    }
?>
```

专家点评

实现数据类型的转换可以采用在变量前面加上一个小括号，并把目标数据类型写在小括号中即可。另外，也可以通过 settype()函数来设置变量的类型。其语法如下：

```
Bool settype ( mixed var, string type )
```





settype()函数的参数说明如表 4.4 所示。

表 4.4　settype()函数的参数说明

参　数	说　明
var	必要参数。输入的变量
type	必要参数。变量的类型。可能值为： “boolean”（或为“bool”，从php4.2.0起） “integer”（或为“int”，从php4.2.0起） “float”（只在 php4.2.0之后可以应用，对于旧版本中应用的“double”现已停用） “string” “array” “object” “null”（从 php4.2.0起）

问题 130　PHP 如何连接 SQL Server 数据库?

问题阐述

PHP 经常与 MySQL 数据库合作，但有时也会与 SQL Server 数据库合作，那么在 PHP 中如何连接 SQL Server 数据库呢？

专家解答

PHP 连接 SQL Server 数据库的方法有两种：一是通过 ADO 方式连接；二是通过 mssql_connect()函数连接。下面分别介绍这两种连接方式。

1. 通过 ADO 方式连接 SQL Server 数据库

利用 ADO 方式建立与 SQL Server 数据库的连接，与使用 ADO 方式建立与 Access 数据库的连接的方法类似。建立连接是在本实例的 conn.php 文件中实现的，代码如下：

```
<?php
$conn=new com("adodb.connection");
$connstr="provider=sqloledb;data source=localhost;uid=sa;pwd=;database=db_online";
$conn->open($connstr);
?>
```

2. 通过 mssql_connect()函数连接 SQL Server 数据库

通过 mssql_connect()函数建立与 SQL Server 数据库的连接首先应该配置 php.ini 文件，将 extension=php_mssql.dll 前的注释去掉，然后重新启动 Apache 服务器。通过 mssql_connect()函数建立与数据库连接的代码如下：

```
<?php
$conn=mssql_connect("localhost","sa","");                //建立与 SQL Server 数据库的连接
mssql_select_db("db_online ",$conn);                     //选择数据库
?>
```

例如，下面使用 mssql_connect()函数连接 SQL Server 数据库，连接结果如图 4.56 所示。



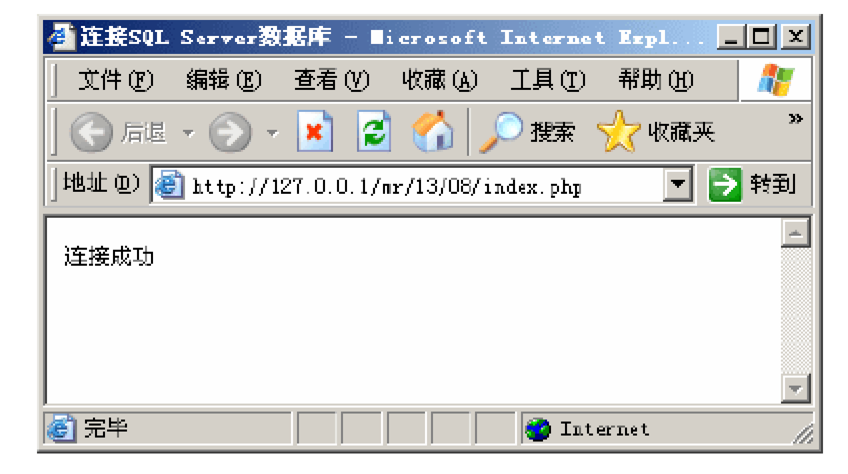



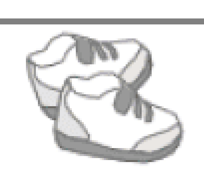
图 4.56 连接 SQL Server 数据库

连接 SQL Server 数据库的代码如下：

```
<?php
$conn=mssql_connect("LLL-SC08Z7OW0V9","sa","");    //建立与 SQL Server 数据库的连接
$select=mssql_select_db("master",$conn);            //选择数据库
if($select){
    echo"连接成功";
}else{
    echo"连接失败";
}
?>
```



日期和时间处理



问题 131　如何获得当前日期？



问题阐述

系统的当前时间受时区限制。默认情况下，系统的当前时间是格林威治时间。那么如何获得当前日期呢？

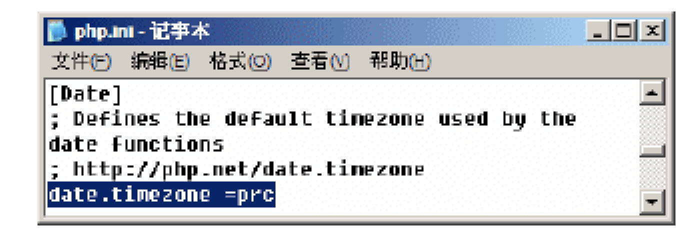
专家解答

因为在 PHP 语言中，日期、时间函数依赖于服务器的地区设置，而 PHP 默认设置的是标准的格林威治时间（即采用的是零时区），所以，如果用户没有对 PHP 的时区进行设置，那么用户使用日期、时间函数获取的将是英国伦敦本地时间（即零时区的时间）。这也就是我们为什么要对 PHP 的时区进行设置的原因，因为如果不设置正确的时区，那么 PHP 的日期、时间函数就获取不到正确的当地时间。例如，以东八区为例，如果当地使用的是北京时间，那么如果没有对 PHP 的时区进行设置，那么获取的时间就将比当地的北京时间少 8 个小时。

更改 PHP 语言中的时区设置有两种方法：

（1）在 php.ini 文件中，定位到[date]下的“;date.timezone =”选项，去掉前面的分号，并设置它的值为当地所在时区使用的时间。修改内容如图 5.1 所示。

例如，如果当地所在时区为东八区，那么就可以设置“date.timezone =”的值为：PRC、Asia/Hong_Kong、Asia/Shanghai（上海）或 Asia/Urumqi（乌鲁木齐）等。这些都是东八区的时间。



php.ini - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

```
[Date]
; Defines the default timezone used by the
date functions
; http://php.net/date.timezone
date.timezone =prc
```

图 5.1　设置 PHP 的时区

设置完成后，保存文件，重新启动 Apache 服务器。

（2）在应用程序中，在日期、时间函数之前使用 date_default_timezone_set()函数就可以完成对时区的设置。date_default_timezone_set()函数的语法如下：

```
date_default_timezone_set(timezone);
```

参数 timezone 为 PHP 可识别的时区名称。如果时区名称 PHP 无法识别，则系统采用 UTC 时区。

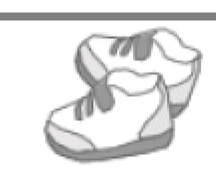
例如，设置北京时间可以使用的时区包括：PRC（中华人民共和国）、Asia/Chongqing（重庆），Asia/Shanghai（上海）或者 Asia/Urumqi（乌鲁木齐），这几个时区名称是等效的。

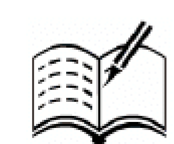
获得服务器当前日期的最简单方法是使用 time 函数。time 返回当前日期的 UNIX 时间戳。可以使用 data 函数格式化日期使其易于理解。

问题 132　如何知道是星期几？

问题阐述

我们按照星期来安排生活。谈论日期时，经常使用像“下个星期二”或“上个星期三”之类的短语。对于我们来说，理解这种方式的日期要比普通日期容易很多，因为不需要计算这个日期是不是下个星期二。因此，对于以一种很随意的格式给定的日期（如 2011 年 6 月 30 日），如何才能确定这个日期是代表一个星期中的哪一天呢？





专家解答

将日期传递给 strtotime 函数，该函数具有理解日期并将其自动转换为 UNIX 时间戳的无限能力，这样就可以使用 data 函数和 l（小写的 L）占位符转换这个时间戳了。

strtotime()函数将任何英文文本的日期时间描述解析为 UNIX 时间戳。语法如下：

```
int strtotime ( string time [, int now] )
```

strtotime()函数接受一个包含英语日期格式的字符串并尝试将其解析为 UNIX 时间戳（自 January 1 1970 00:00:00 GMT 起的秒数），其值相对于 now 参数给出的时间，如果没有提供此参数则用系统当前时间。

如果参数 time 的格式是绝对时间，则 now 参数不起作用；如果参数 time 的格式是相对时间，那么其对应的时间由参数 now 来提供。

如果解析成功，则返回时间戳，否则返回 False。在 PHP 5.1.0 之前本函数在失败时返回-1。

下面应用 strtotime 函数和 date 函数获取指定时间的周日期格式。

```
<?php
$time=strtotime("Jul 7st 2011");
$weekday=date("l",$time);
echo $weekday;
?>
```

问题 133　如何获得一个月份的天数？

问题阐述

获得一个月份的天数是很常见的任务，特别是在编写基于日期的应用程序时，如日历等。那么如何应用日期函数获取一个月份的天数呢？

专家解答

使用 strtotime 函数、date 函数以及 t 占位符可以很容易地获得这种信息。其方法如下：

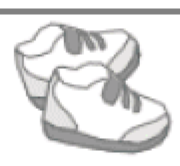
```
<?php
$time=strtotime("Jul");                //返回时间戳
$weekday=date("t",$time);              //通过 date 函数输出时间，定义格式为 t
echo $weekday;
?>
```

问题 134　如何在 MySQL 中存储日期？

问题阐述

可读的日期有很多种格式，能够适合很多情况。然而，这些格式都不是存储日期的最好方式。在 MySQL 中存储日期的最简单方法是按照其在网页上显示的方式存储，例如“8th Marth 2011”，采用这种方式非常不理想。

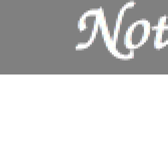
因为，在 MySQL 中运行的 SQL 语句的 WHERE 子句不允许出现下面的情况：



```
SELECT * FROM table WHERE date >'14th February 2011'
```

14th February 2011 不是一个日期值，而只是一个字符串表示的日期，它无法像日期一样操作或比较，除非它被转换为日期值。如果用户以字符串方式存储日期，将需要不停地将它们在字符串和日期类型的值之间来回转换，这样做是十分让人头疼的。那么应该采用什么样的格式来存储日期呢？





专家解答

最理想的存储日期信息的方法是使用 MySQL 时间戳。为了获得服务器本地时区的当前时间，可以使用 now 或 CURRENT_TIMESTAMP 函数，也可以使用 UTC_TIMESTAMP 函数获得 UTC 时区时间戳。

通过 MySQL 日期时间函数来获取时间，完全符合 MySQL 数据库存储的要求，是最为理想的方法。例如，在下面的语句中分别应用 MySQL 中的 CURRENT_TIMESTAMP、now 和 UTC_TIMESTAMP 函数获取当前日期时间，如图 5.2 所示。

```
mysql> select current_timestamp();
+---------------------+
| current_timestamp() |
+---------------------+
| 2011-07-09 16:35:27 |
+---------------------+
1 row in set (0.02 sec)

mysql> select now();
+---------------------+
| now()               |
+---------------------+
| 2011-07-09 16:36:32 |
+---------------------+
1 row in set (0.00 sec)

mysql> select utc_timestamp();
+---------------------+
| utc_timestamp()     |
+---------------------+
| 2011-07-09 08:37:37 |
+---------------------+
1 row in set (0.00 sec)
```

图 5.2　应用 MySQL 函数获取当前日期时间

专家点评

MySQL 时间戳要比 UNIX 时间戳简单一些，其通用形式是“YYYY-MM-DD HH:MM:SS”，通常存储在类型为 DATETIME（不要和 DATE 及 TIME 弄混，它们分别存储“YYYY-MM-DD”和“HH:MM:SS”形式的值）的列中。

这种形式的时间戳适用于简单的排序和比较操作，而且具有易读性的优点，同时还具有可以预计的长度，这使得它们比较容易验证。

可以通过 MySQL 的 DATETIME 列类型利用许多 MySQL 自身的日期和时间函数，如果需要，DATETIME 列类型也很容易转换为 UNIX 时间戳。

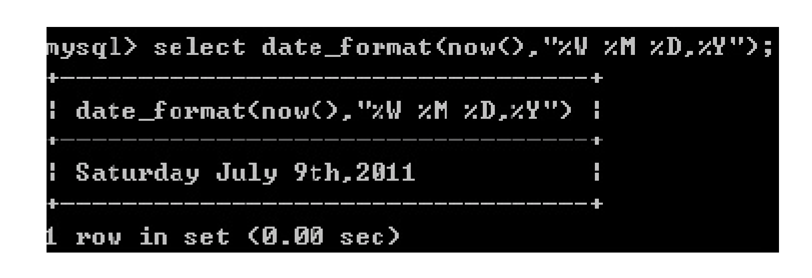
问题 135　如何使用 MySQL 时间戳？

问题阐述

尽管 MySQL 时间戳具有可读性，但却并不是完美的，例如，在网页页面上输出 2011-07-09 16:50:30，就显得过于拖沓，相反使用“February 8th，2011 at 7 p.m.”就显得工整时尚了。那么这种格式应该如何实现呢？

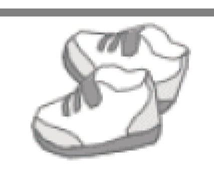
专家解答

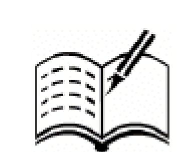
MySQL 像 PHP 一样具有一个日期格式化函数 DATE_FORMAT，使用该函数时，只需要像下面这样传递给它一个格式字符串和一个时间戳即可。



```
mysql> select date_format(now(),"%W %M %D,%Y");
+----------------------------------+
| date_format(now(),"%W %M %D,%Y") |
+----------------------------------+
| Saturday July 9th,2011           |
+----------------------------------+
1 row in set (0.00 sec)
```

图 5.3　应用 DATE_FORMAT 函数转换时间





专家点评

DATE_FORMAT 很像 PHP 的 date 函数，也使用包含定义格式的修饰符的格式字符串。表 5.1 中列出了 DATE_FORMAT 函数的常用修饰符。

表 5.1　DATE_FORMAT 函数的常用修饰符

参　数	说　明
%a	星期的简写（Sun~Sat）
%b	月份的简写（Jam~Dec）
%c	月，数字（1~12）
%d	有英语前缀的月份的日期
%D	月份中的天数，数字（00~31）
%e	月份中的天数，数字（00~31）
%I	12小时制的十进制小时数（范围从00~12）
%j	年份中的第几天，十进制数（范围从001~366）
%m	十进制月份（范围从01~12）
%p	根据给定的时间值为am或pm，或者当前区域设置中的相应字符串
%r	用a.m和p.m符号的时间
$S	十进制秒数
%T	当前时间，和%H:%M:%S一样
%W	本年的第几周数，从第一周的第一个星期一作为第一天开始
%w	星期中的第几天，星期天为0
%y	没有世纪数的十进制年份（范围从00~99）
%Y	包括世纪数的十进制年份
%%	文字上的%字符

问题 136　如何实现倒计时？

问题阐述

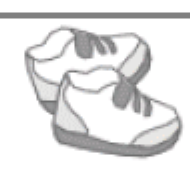
在生活中，我们经常会看到对期待某一日子或某一时间的到来实行倒计时计算。例如，2012 年伦敦奥运会、每年春节晚会新的一年到来前等。那么，在程序中如何实现倒计时呢？

专家解答

倒计时的原理非常简单，就是用一个固定的时间减去当前的时间，所得到的就是剩余时间。要完成时间的加减操作，与比较大小类似，都要将时间转换成时间戳，然后才能计算，最后再将时间戳转换成日期输出。

将时间转换成时间戳自然应用的是 strtotime()函数，其方法如下：

```
<?php
    $time1 = strtotime(date( "Y-m-d"));              //当前的系统时间
    $time2 = strtotime("2012-1-1");                  //2012 年元旦
    $sub2 = ceil(($time2 - $time1) / 86400);         //(60 秒*60 分*24 小时)秒/天
```



```
    echo "距离 2012 年元旦还有<font color=red>$sub2 </font>天!!!";
?>
```



Note

问题 137　如何计算程序的运行时间？

问题阐述

在百度中，当我们执行一个查询操作时，获取到查询结果后，页面中就会出现一行文字，提示用户根据关键字搜索到多少个结果以及搜索所用的时间。这个时间就是程序在执行这个搜索时所用的时间，那么它是如何实现的呢？

专家解答

计算程序运行时间的原理：首先，在执行查询操作之前定义一个时间，将其精确到微秒，然后，执行查询操作，接着，在查询功能执行完毕后，再次获取一个时间，同样精确到微秒，最后，应用后获取的时间减去最初获取的时间，就是本次查询所用的时间。其具体实现方法如下：

创建 index.php 文件，首先定义 run_time()方法，然后调用 run_time()方法，获取当前时间戳，接着运行程序，最后再次调用 run_time()函数，获取系统当前时间，并且输出差值，获取程序的运行时间，其代码如下：

```
<?php
/*      声明 run_time 函数
通过 microtime()获取当前时间的微秒数和时间戳
应用 explode()函数将 microtime()函数返回的字符串进行分隔，返回一个数组
包括两个元素，一个元素是当前时间的微秒数，另一个元素是当前时间的时间戳
应用 list()函数将 explode()函数返回的数组值赋给指定的变量
最后获取两个变量值的和
*/
function run_time(){
list($msec, $sec) = explode(" ", microtime());          //使用 explode()函数返回两个变量
return ((float)$msec + (float)$sec);                     //返回两个变量的和
}
$start_time = run_time();                                //第一次运行 run_time()函数
/*      运行 PHP 代码段      */
    $time1 = strtotime(date( "Y-m-d"));                  //当前的系统时间
     $time2 = strtotime("2012-10-1");                    //2012 国庆节
     $sub2 = ceil(($time2 - $time1) / 86400);            //(60 秒*60 分*24 小时)秒/天
  echo "距离 2012 年国庆节还有<font color=red>$sub2 </font>天!!!";
/*      *******************       */
$end_time = run_time();                                  //再次运行 run_time()函数
?>
<!--  输出差值，获取程序的运行时间  -->
倒计时程序的运行时间为<font color=blue> <?php echo ($end_time - $start_time); ?> </font>秒
?>
```

专家点评

在计算程序的运行时间过程中，应用 strtotime()函数获取时间戳，实现倒计时功能，应用 microtime()函数将时间精确到微秒，并计算出程序的运行时间。



问题 138 如何获取当前系统时间？

问题阐述

如何通过程序获取当前系统的准确时间呢？

专家解答

1. 解决思路

获取系统当前时间可以使用 date()函数，该函数用于格式化一个本地时间/日期，语法格式如下：

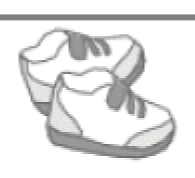
```
string date ( string format , int timestamp)
```

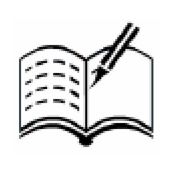
该函数返回将参数 timestamp 按照指定格式字符串而产生的字符串，其中的参数 timestamp 是可选的，默认值为 time()，即如果没有给出时间戳则使用本地当前时间。

date()函数的参数 format 的格式化选项如表 5.2 所示。

表 5.2 参数 format 的格式化选项

参 数	说 明
a	小写的上午和下午值，返回值为am或pm
A	大写的上午和下午值，返回值为AM或PM
B	Swatch Internet标准时，返回值为000~999
d	月份中的第几天，有前导零的2位数字，返回值为01~31
D	星期中的第几天，文本格式，3个字母，返回值为Mon~Sun
F	月份，完整的文本格式，返回值为January~December
g	小时，12小时格式，没有前导零，返回值为1~12
G	小时，24小时格式，没有前导零，返回值为0~23
h	小时，12小时格式，有前导零，返回值为01~12
H	小时，24小时格式，有前导零，返回值为00~23
i	有前导零的分钟数，返回值为00~59
I	判断是否为夏令时，返回值如果是夏令时，为1，否则为0
j	月份中的第几天，没有前导零，返回值为1~31
l	星期数，完整的文本格式，返回值为Sunday~Saturday
L	判断是否为闰年，返回值如果是闰年为1，否则为0
m	数字表示的月份，有前导零，返回值为01~12
M	3个字母缩写表示的月份，返回值为Jan~Dec
n	数字表示的月份，没有前导零，返回值为1~12
o	与格林威治时间相差的小时数，如0200
r	RFC 822 格式的日期，如Thu，21 Dec 2000 16：01：07 +0200
s	秒数，有前导零，返回值为00~59
S	每月天数后面的英文后缀，两个字符，如st、nd、rd或th。可以和j一起使用





Note

续表

参　数	说　明
t	指定月份所应有的天数
T	本机所在的时区
U	从UNIX纪元（January 1 1970 00:00:00 GMT）开始至今的秒数
w	星期中的第几天，数字表示，返回值为0~6
W	ISO-8601格式年份中的第几周，每周从星期一开始
Y	4位数字完整表示的年份，返回值如1998、2008
y	两位数字表示的年份，返回值如88或08
z	年份中的第几天，返回值为0~366
Z	时差偏移量的秒数。UTC西边的时区偏移量总是负的，而UTC东边的时区偏移量总是正的，返回值为-43200~43200

说明：

有效的时间戳典型范围是格林威治时间 1901 年 12 月 13 日 20:45:54 到 2038 年 1 月 19 日 03:14:07（此范围符合 32 位有符号整数的最小值和最大值）。在 Windows 系统中此范围限制为从 1970 年 1 月 1 日到 2038 年 1 月 19 日。

2. 应用示例

本示例主要应用 date()函数来获取系统的当前时间，运行结果如图 5.4 所示。



图 5.4　获取系统当前时间

应用 date()函数获取系统的当前时间，时间的格式为“yyyy-mm-dd hh:ii:ss”。其代码如下：

```
<?php
date_default_timezone_set("Asia/Hong_Kong");
$data=date("Y-m-d h:i:s");
echo $data;
?>
```

专家点评

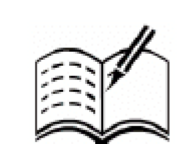
由于在 PHP 语言中默认设置的是标准的格林威治时间（即采用的是零时区），所以要获取本地当前的时间必须更改 PHP 语言中的时区设置，有两种方法：（1）修改 php.ini 文件中的设置，找到[date]下的“;date.timezone =”选项，将该项修改为“date.timezone =Asia/Hong_Kong”，然后重新启动 Apache 服务器。（2）在应用程序中，在使用时间/日期函数之前添加函数“date_default_timezone_set("Asia/Hong_Kong");”即可。

问题 139　如何实现网页提醒功能？

问题阐述

闹钟是生活中经常使用的一个小工具，它会在指定时间叫你起床，那么我们是否可以在网页中制作一个闹钟呢？当系统时间运行到某一时刻时，为浏览者提供一个温馨的提示信息呢？





例如，当母亲节来临之时，访问网站的用户会看到一个温馨的提示，“母亲节到了，记得向您的母亲道一声辛苦了”。

专家解答

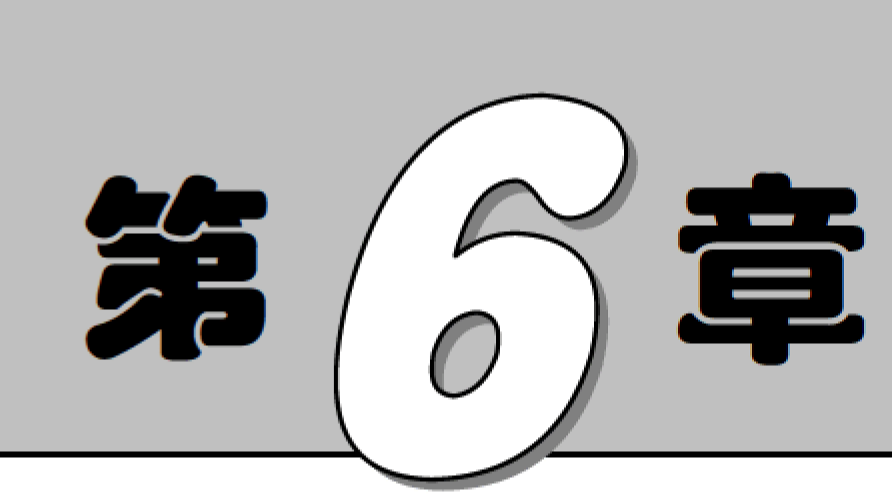
网页提醒功能，通过判断系统中当前的时间与指定的某个时间或时间段是否相同，如果相同，系统就给出一个对应的提示信息，提示用户该做什么（如每天工作记录的上传、每月经验技巧的提交或者员工生日的提醒等）。

上述比较可以有两种方法，第一种方法是将时间转换为时间戳，然后进行比较。例如，当时间运行到 9 月 18 日时，提醒千千万万的中华同胞们勿忘国耻。

```
<?php
$time1 = strtotime(date("Y-m-d"));                    //当前的系统时间，获取月和天的时间戳
$time2 = strtotime(date("Y")."-09-18");               //设置时间 9 月 18 日的时间戳
if($time1==$time2){                                   //判断两个时间戳是否相同
  echo "<script>alert('勿忘国耻！');window.location.href='index.php';</script>";  //给出提示信息
}
?>
```

第二种方法是应用字符串函数 substr()对系统当前时间进行截取，将截取后的内容与指定的时间进行比较，如果相同则给出提示信息。例如，当时间运行到 7 月 1 日时，提醒今天是党的生日。

```
<?php
$time = substr(date("Y-m-d"),5,5);                    //截取系统当前时间
if($time=="08-18"){                                   //判断时间是否相同
  echo "<script>alert('生日快乐！');window.location.href='index.php';</script>";  //给出提示信息
}
?>
```

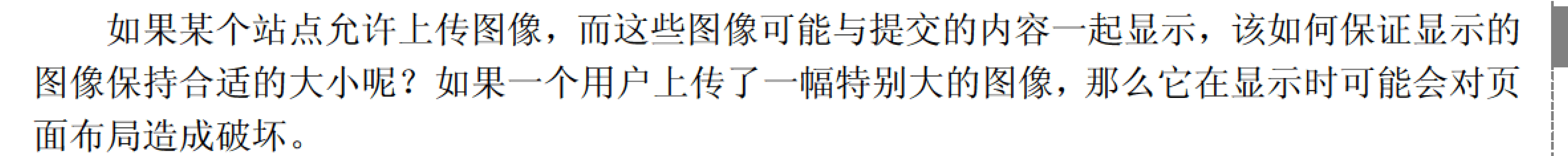


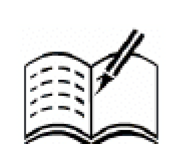
图形图像处理

- 问题 140　如何创建缩略图?
- 问题 141　如何防止图像被盗链?
- 问题 142　在 PHP 中如何激活和屏蔽 GD2 函数库?
- 问题 143　如何在照片上写文字?
- 问题 144　在 GD2 函数库中，如何输出中文字符串?
- 问题 145　如何应用 GD2 函数为图片添加图像水印?
- 问题 146　如何应用 GD2 函数生成带有干扰的图像验证码?
- 问题 147　如何通过图像显示密码安全强度?
- 问题 148　如何从图像中提取 EXIF 信息?
- 问题 149　如何应用 JQuery 和 Jpgraph 制作动态折线图?
- 问题 150　如何通过 Jpgraph 类库输出中文字符串?
- 问题 151　如何制作数字图像验证码?
- 问题 152　如何制作中文图像验证码?
- 问题 153　如何实现验证码在当前页验证?
- 问题 154　如何任意调整图片的大小?



问题 140　如何创建缩略图？





问题阐述

如果某个站点允许上传图像，而这些图像可能与提交的内容一起显示，该如何保证显示的图像保持合适的大小呢？如果一个用户上传了一幅特别大的图像，那么它在显示时可能会对页面布局造成破坏。

专家解答

这个问题的一种解决方案是创建缩略图，可以保证显示的图像永远不会超出某个高度和宽度值。创建一个基本的缩略图需要 5 个步骤：

（1）将源图像装载到一个 PHP 变量中。

（2）确定原有图像的高度和宽度。

（3）创建一个具有正确尺寸的空白缩略图。

（4）复制原有图像到空白缩略图。

（5）使用正确的内容类型显示缩略图。

下面为一幅 JPEG 格式的照片创建一个缩略图。

（1）设置源图像的路径和期望的宽度和高度像素值。

```
<?php
$sourceImage = 'images/1.jpg';
$thumbWidth = 200;
$thumbHeight = 200;
```

（2）使用 imagecreatefromjpeg 创建一个 jpeg 图像，通过 getimagesize 函数返回图像的宽度和高度。

```
$original = imagecreatefromjpeg($sourceImage);
$dims = getimagesize($sourceImage);
```

（3）使用 imagecreatetruecolor 函数创建一个空白图像（以 PHP 变量$thumb 的形式存在内存中），缩略图将被放置在其中。

```
$thumb = imagecreatetruecolor($thumbWidth,$thumbHeight);
```

（4）通过 imagecopyresampled 函数生成缩略图。

```
imagecopyresampled( $thumb, $original, 0, 0, 0, 0,$thumbWidth, $thumbHeight, $dims[0], $dims[1] );
```

imagecopyresampled 函数将图像的一个调整大小后的版本放置到空白缩略图中，以这种方法重新取样能够确保图像平滑地调整大小。

参数$thunb 定义目的图像；参数$original 定义源图像。后面的 4 个参数代表目的图像和源图像部分的 X 和 Y 坐标，坐标系原点位于左上角。

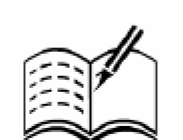
（5）通过 header()函数定义内容类型头（Content-type: image/jpeg），使用 imagejpeg()函数输出完成的缩略图：

```
header( "Content-type: image/jpeg" );
```



```
imagejpeg( $thumb );
?>
```

运行结效果如图 6.1 所示。



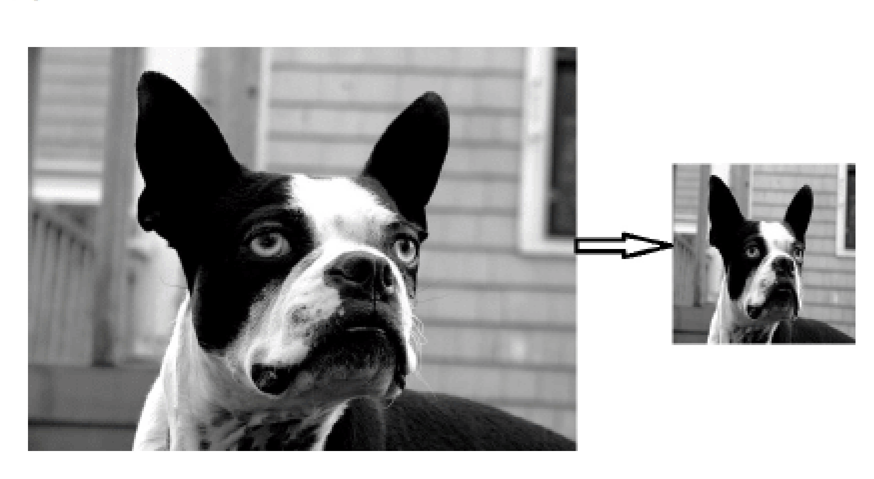


图 6.1　缩略图

问题 141　如何防止图像被盗链?

问题阐述

网站的图片被盗链是一件让人非常郁闷的事情。盗链不仅盗用图片，更直接的问题是用户在下载盗链人网站上的图片时，会给服务器带来压力，导致日志中的访问记录暴涨，而带宽被耗尽。那么如何才能防止图片被盗链呢?

专家解答

要防止网站图片被盗链，那么就有必要了解盗链的手段，基本上可以分为两种：

☑ 第一种：盗链者直接使用图片链接引用网站上的图片。

防止这种盗链情况的发生有两种可行的方法：通过 Apache 服务器进行控制和使用 SESSION 变量在程序中进行控制。

☑ 第二种：直接从网站上下载图片并且复制使用。

防止图像盗链的方法主要有两种解决方案，一种使用 Apache 的 mod_rewrite.so 扩展，另一种使用 PHP 的 SESSION 变量。

1. Apache 防盗技术

Apache 防盗技术的原理与 PHP 伪静态技术的原理是相同的，都需要应用 Apache 的 mod_rewrite.so 模组。Apache 服务器的配置文件 httpd.conf 的修改方法如下：

（1）打开 httpd.conf 文件，定位到如下位置：

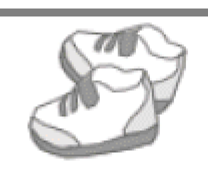
```
#LoadModule rewrite_module modules/mod_rewrite.so
```

将该项前面的“#”去掉，启动该项。

（2）查找 httpd.conf 文件，找到其中的“AllowOverride”项，将它的值修改为 All。保存并重新启动 Apache 服务器，使修改生效。

（3）在项目根目录下创建.htaccess 文件，定义防止网站图片被盗链的方法。.htaccess 文件的代码如下：

```
SetEnvIfNoCase Referer "^http://192.168.1.59/" local_ref=1
<FilesMatch ".(gif|jpg)">
Order Allow,Deny
```



```
Allow from env=local_ref
</FilesMatch>
```

☑ Referer 字段：当 Apache 处理一个请求时，将检测头信息中的 Referer 字段，并且设置环境变量 local_ref 为 1，如果请求从本身的网站地址开始，即是本网站的一个页面。
☑ ^http://192.168.1.59/：是一个正则表达式，为了设置环境变量，Referer 值必须匹配它。
☑ "NoCase"指令：定义正则表达式的值忽略字符串的大小写。
☑ Order Allow,Deny：设置 Apache 对当前的请求，将执行列表中的 Allow 指令，然后重复进行 Deny 指令。
☑ local_ref：这将使设置了 local_ref 环境变量（无论什么值）的请求通过。任何其他的请求将被拒绝，因为它们不符合 Allow 的条件并且默认是拒绝访问的。



2. SESSION 变量防盗链

SESSION 变量防盗链技术的原理是判断图片访问者的权限，如果有权限则可以访问，否则就不能够访问。具体实施就是，首先定义一个 SESSION 变量，然后使用另一个脚本生成图像，并且在这个脚本中判断这个 SESSION 变量是否存在，如果存在则可以访问图像。例如，下面是一个显示一幅图像的简单页面，图像的源 URL 是一个 PHP 脚本，确保只有站点的参观者才能够看到图像。

```
<?php
session_start();                                   //初始化 SESSION 变量
$_SESSION['viewimages']=true;                      //定义 SESSION 变量，并为其赋值
?>
<tr>
        <td colspan="2">
            <img src="getimage.php?img=bg3_01.jpg" width="267" height="31" alt=""></td>
        <td>
            <img src="getimage.php?img=bg3_02.jpg" width="536" height="31" alt=""></td>
    </tr>
```

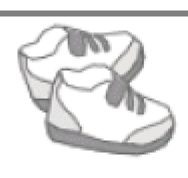
注意在上面的代码中登记了一个称为 viewimages 的 SESSION 变量，而<img>标记的 src 属性中显示的是 getimage.php?img=bg3_01.jpg。

下面是脚本 getimage.php 的代码。首先，检查 SESSION 变量 viewimage，看其是否被设置为 true：

```
<?php
session_start();                                   //初始化 SESSION 变量
if (isset($_SESSION['viewimages']) && $_SESSION['viewimages'] == true) {     //判断 SESSION 变
                                                                              量是否为真
```

如果 SESSION 变量 viewimages 被设置为 true，那么将会检查$_GET ['img']变量中提供的图像是否存在，请求有效，因此输出响应头和图像：

```
$dims = getimagesize ( 'images/' . $_GET ['img'] );        //获取超链接传递图片的信息
        header ( 'Content-Disposition: inline; filename=' . $_GET ['img'] );
        header ( 'Content-Type:' . $dims ['mime'] );
        header ( 'Content-Length:' . filesize ( 'images/' . $_GET ['img'] ) );
        readfile ( 'images/' . $_GET ['img'] );            //完成对图片的读取操作
    }
```





Note

注意，这段脚本还使用了 getimagesize 函数获取图像的尺寸。

最后，通过返回 HTTP 状态“404Not Found”来处理错误情况：

```
else {
        header ( "HTTP/1.1 404 NOT Found" );
        header ( "Content-Type: text/plain" );
        echo "图像不存在！ \n";
    }
} else {
    header ( "HTTP/1.1 404 NOT Found" );
    header ( "Content-Type: text/plain" );
    echo "这是一个被保护的图像不可盗链！ \n";

}
```

这种保护能够阻止大多数盗链者。

专家点评

使用 PHP 的 SESSION 变量防盗链，这种方法防止图片被盗链操作起来相对通过 Apache 来防止图片被盗链要麻烦一些，但是如果在购买的虚拟主机中使用，那么必须考虑虚拟主机的 Apache 配置是否支持 mod_rewrite.so 模组，而使用 SESSION 变量来防止图片被盗链，则不必考虑这个问题，因为它完全是在用户的程序中运行的。所以，具体使用哪种方法来防止网站图片被盗链，应根据用户的实际情况而定。

问题 142　在 PHP 中如何激活和屏蔽 GD2 函数库？

问题阐述

在 PHP 中如何激活和屏蔽 GD2 函数库？

专家解答

GD 库在 PHP 5 中是被默认安装的，但是要想激活 GD 库，则必须修改 php.ini 文件。将该文件中的“;extension=php_gd2.dll”选项前的分号“;”去掉，保存文件并重新启动 Apache 服务器即可生效。如果要屏蔽对 GD2 函数库的支持，则在“;extension=php_gd2.dll”选项前加上分号即可。

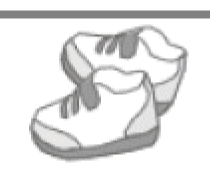
问题 143　如何在照片上写文字？

问题阐述

通过 PHP 可以输出照片，那么如何在照片上写文字呢？

专家解答

在 GD2 函数库中，既可以绘制英文字符串，也可以绘制中文字符串。绘制英文字符串应



用 imagestring()函数。其语法如下：

```
bool imagestring ( resource image, int font, int x, int y, string s, int col )
```

imagestring()函数用 col 颜色将字符串 s 绘制到 image 所代表的图像的（x，y）坐标处（这是字符串左上角坐标，整幅图像的左上角为 0，0）。如果 font 是 1、2、3、4 或 5，则使用内置字体。

通过 imagestring()函数水平地绘制一行字符串“I Like MingRiKeJi”。具体步骤如下：

首先载入一个 jpg 图像，然后通过 imagestring()函数在图片中绘制一行英文字符串。最后输出图像并且销毁图像资源。其代码如下：

```
<?php
header("Content-Type:text/html;charset=utf-8");        //设置页面的编码风格
header("Content-Type:image/jpeg");                     //告知浏览器输出的是一张图片
$img=imagecreatefromjpeg("P1020110.jpg");              //载入照片
$textcolor=imagecolorallocate($img,0,0,0);             //设置字体颜色为蓝色，值为 RGB 颜色值
imagestring($img,5,80,30,"I Like MingRiKeJi",$textcolor);        //书写英文字符
imagejpeg($img);                                                 //输出图像
magedestroy($img);                                               //销毁图像
?>
```

运行结果如图 6.2 所示。



图 6.2　绘制英文字符串

问题 144　在 GD2 函数库中，如何输出中文字符串？

问题阐述

在 GD2 函数库中，如何输出中文字符串？

专家解答

PHP 中的 GD 库支持中文，但必须要以 UTF-8 格式的参数来进行传递，如果使用 imageString()函数直接绘制中文字符串就会显示乱码，这是因为 GD2 函数库对中文只能接收 UTF-8 编码格式，并且默认使用了英文的字体，所以要输出中文字符串，就必须对中文字符串进行转码，并设置中文字符使用的字体；否则，输出的只能是乱码。

PHP 在图像中添加中文字符串应用的是 imagettftext ()函数，其语法如下：

```
array imagettftext ( resource image, float size, float angle, int x, int y, int color, string fontfile, string text )
```

imagettftext ()函数的参数说明如表 6.1 所示。

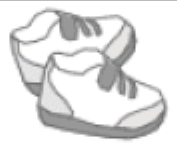



表 6.1　imagettftext ()函数的参数说明

参　　数	说　　明
image	图像资源
size	字体大小。根据GD版本不同，应以像素大小指定（GD1）或点大小（GD2）
angle	字体的角度，顺时针计算，0度为水平，也就是3点钟的方向（由左到右），90° 则为由下到上的文字
x	文字的x坐标值。设定第一个字符的基本点
y	文字的y坐标值。设定字体基线的位置，不是字符的最底端
color	文字的颜色
fontfile	字体的文件名称，也可以是远端的文件
text	字符串内容

PHP 中通过 GD2 函数对 JPG 格式的照片进行操作，应用 imagecreatefromjpeg()函数，从 JPEG 文件或 URL 新建一图像。其语法如下：

```
resource imagecreatefromjpeg ( string filename )
```

参数 filename 可以是本地文件，也可以是网络的 URL 地址。返回值为 JPEG 的文件代码，可供其他函数使用。该函数在失败时返回一个空字符串，并且输出一条错误信息。

应用示例

下面通过 GD2 函数在图片中输出中文字符串。其具体代码如下：

```
<?php
header("content-type:image/jpeg");                    //定义输出为图像类型
$im=imagecreatefromjpeg("images/P1020494.JPG"); //载入照片
$textcolor=imagecolorallocate($im,25,25,136);         //设置字体颜色为蓝色，值为 RGB 颜色值
$font="Font/mzd.ttf";                                 //定义字体
$to="落霞与孤鹜齐飞";
imagettftext($im,80,0,200,200,$textcolor,$font,$to);  //写 TTF 文字到图中
$to="秋水共长天一色";
imagettftext($im,80,0,300,400,$textcolor,$font,$to);  //写 TTF 文字到图中
imagejpeg($im);                                       //建立 JPEG 图形
imagedestroy($im);                                    //结束图形，释放内存空间
?>
```

运行结果如图 6.3 所示。



图 6.3　在照片上输出中文字符串

专家点评

由于 imagettftext()函数只支持 UTF-8 编码，所以在创建文件时必须也要使用 UTF-8 编码格式，这样才能保证中文字符串的正常输出。但是，如果页面本身使用的是 gb2312 编码格式，那么就必须应用 iconv()函数对向图片中添加的中文字符串进行编码格式的转换，由 gb2312 编码转换为 UTF-8 编码。



问题 145　如何应用 GD2 函数为图片添加图像水印？

问题阐述

如何为上传图片添加一个图像水印？

专家解答

使用图片作为水印的前提是该图片的背景必须是透明的，否则输出的效果将很不理想。图片水印添加的关键是 getimagesize()和 imagecopy()函数。应用 getimagesize()函数获取上传图片和水印图片的大小，通过 imagecopy()函数完成图片水印的添加。

imagecopy()函数将图像复制到指定的另外一个图像中。其语法如下：

```
bool imagecopy ( resource dst_im, resource src_im, int dst_x, int dst_y, int src_x, int src_y, int src_w, int src_h )
```

将 src_im 图像中坐标从 src_x，src_y 开始，宽度为 src_w，高度为 src_h 的一部分复制到 dst_im 图像中坐标为 dst_x 和 dst_y 的位置上。

应用示例

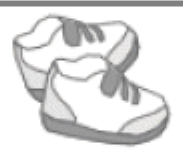
在本示例中，整合图像的创建、操作、输出和销毁函数，创建 AddWaterPress 类，定义 getExtendsName()方法，获取上传图片的文件后缀；定义 getImageRes()方法，根据上传文件的后缀创建新图像；定义 outputImage()方法，输出图像；定义 add()方法，向指定的图片中添加图像水印。其具体实现过程如下：

（1）创建 index.php 文件。首先，创建一个表单，完成上传图片的提交操作。然后，在本页中输出存储在服务器指定文件夹下的图片。最后，获取表单中提交的图片数据，应用 move_uploaded_file()函数完成图片的上传操作，并且实例化存储在 AddWaterPress.php 文件中的 AddWaterPress 类，调用 add()方法为指定的图片添加文字水印。其关键代码如下：

```
if(move_uploaded_file($_FILES["file"]["tmp_name"], $saveDir)){      //执行文件上传操作
require_once 'AddWaterPress.php';                                  //包含添加水印操作的文件
        $addWaterPress = new AddWaterPress();                      //类的实例化
    $addWaterPress->add($saveDir, "吉林省明日科技");                //执行添加方法，传递参数，
                                                                     指定水印文字
        echo "<script>alert('图片添加成功');</script>";
    }
```

（2）在编写 AddWaterPress 类的 add()方法时，应用的是 getimagesize()和 imagecopy()函数完成图像水印的添加操作。Add()方法的代码如下：

```
function add($imageUrl, $watherImageUrl, $x, $y){         //定义添加方法
        $img = @$this->getImageRes($this->getExtendsName($imageUrl), $imageUrl);
                                                           //获取被添加的图像标识
        $img1 = @$this->getImageRes($this->getExtendsName($watherImageUrl), $watherImageUrl);
        $size = getimagesize($imageUrl);                   //获取图像大小
     $size1 = getimagesize($watherImageUrl);               //获取水印图片的大小
        if($x==null && $y==null){                          //判断参数是否为空
            $x1 = ($size[0]-$size1[0])/2;   //根据图像大小数组中返回的值，计算图像的横坐标
            $y1 = ($size[1]-$size1[1])/2;   //根据图像大小数组中返回的值，计算图像的纵坐标
```




```
            }else{
                $x1 = $x;                                   //如果不为空，则直接使用坐标数据
                $y1 = $y;                                   //如果不为空，则直接使用坐标数据
            }
        imagecopy($img, $img1, $x1, $y1, 0, 0, $size1[0], $size1[1]);    //将 img1 的一部分拷贝到 img
                                                                          的指定位置
            //根据图像标识符、后缀和路径，执行 outputImage 方法，输出图像
                $this->outputImage($img, $this->getExtendsName($imageUrl), $imageUrl);
            imagedestroy($img1);                            //销毁图像
        imagedestroy($img);                                 //销毁图像
    }
```

其运行结果如图 6.4 所示。



图 6.4　为图片添加图像水印

专家点评

在将图片作为水印图像时，如果指定的图片背景不是透明的，那么添加的水印图像将与原始图像格格不入，而如果水印图片的背景是透明的，那么添加的水印标记就可以很好地与原始图像融入一体。

问题 146　如何应用 GD2 函数生成带有干扰的图像验证码?

问题阐述

在用户注册时，添加完基本注册信息后，通常会被要求输入验证码，而为了有效增加恶意程序探测验证码的困难程度，常常会在验证码中添加一些干扰元素，如干扰点、干扰线等，这些干扰点或干扰线是如何制作的呢？

专家解答

绘制干扰点主要应用到 imagesetpixel()函数，该函数的语法如下：

```
bool imagesetpixel ( resource image, int x, int y, int color )
```

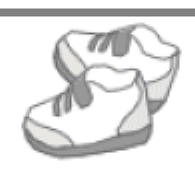
参数说明：

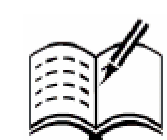
❶ image：图像标识。
❷ x：点的 x 坐标。
❸ y：点的 y 坐标。
❹ color：指定点的颜色。

例如，用 imagesetpixel()函数在图像上绘制 5 个相连接的点，代码如下：

```
$im=imagecreatefromjpeg("bg.jpg");
$color=imagecolorallocate($im,255,255,255);
imagesetpixel ($im,100,80,$color);
imagesetpixel ($im,100,81,$color);
imagesetpixel ($im,100,82,$color);
imagesetpixel ($im,100,83,$color);
imagesetpixel ($im,100,84,$color);
imagejpeg($im);
```







本示例的运行结果如图 6.5 所示。

应用示例

本示例应用 GD2 函数绘制图像验证码，并通过 JavaScript 调用所绘制的验证码显示在用户注册表单中，结果如图 6.6 所示。



图 6.5　imagesetpixel()函数在图像上连续绘制 5 个点

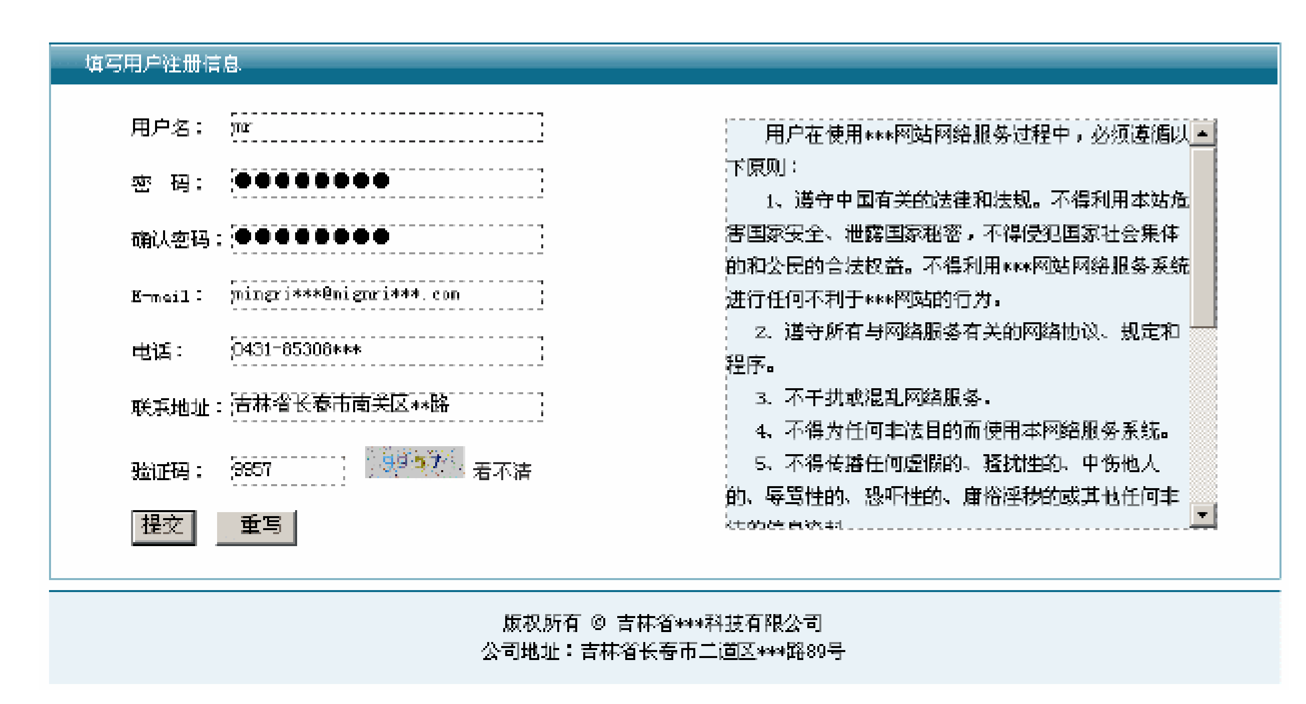


图 6.6　带有干扰线的验证码

通过 JavaScript 脚本生成的 4 位随机数作为验证码的值，并且通过 JavaScript 脚本中的方法完成验证码的二次生成。当用户单击“看不清”超链接时，不刷新页面而重新生成验证码，并将验证码的值赋给隐藏域。其关键代码如下：

```
<script language="javascript">
      var num1=Math.round(Math.random()*10000000);
      var num=num1.toString().substr(0,4);                    //产生 4 位随机数
      document.write("<img name=codeimg src='ValidatorCode.php?code="+num+"'>");
      form1.defValidatorCode.value=num;                       //将随机数赋给隐藏域
function reCode(){                                            //该方法用于重新生成验证码
var num1=Math.round(Math.random()*10000000);
var num=num1.toString().substr(0,4);
document.codeimg.src="ValidatorCode.php?code="+num;
form1.defValidatorCode.value=num;
}
</script>
```

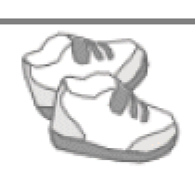
通过 GD2 函数将 JavaScript 脚本生成的 4 位随机数绘制成图像，并且为验证码的图像添加干扰点。其中添加干扰点的代码如下：

```
for ($i = 0; $i < 200; $i ++) {                               //绘制 200 个干扰点
      imagesetpixel($im, rand() % 70, rand() % 30, imagecolorallocate($im, rand(0, 255), rand(0, 255),
rand(0, 255)));
}
```

问题 147　如何通过图像显示密码安全强度？

问题阐述

在注册页设置密码时，常常会有图像显示所设密码的强度，这是如何实现的呢？



专家解答



对密码的安全强度进行判断是在 JavaScript 脚本中应用正则表达式来完成的。其原理如下：

（1）对密码中的值进行判断，如果只包含数字、英文字符串或特殊字符串中的任意一种，那么说明密码安全强度差。

（2）对密码中的值进行判断，如果包含数字、英文字符串或特殊字符串中的任意两种，那么说明密码安全强度中。

（3）对密码中的值进行判断，如果同时包含数字、英文字符串和特殊字符串，那么说明密码安全强度强。

在 JavaScript 脚本中，通过 search()方法调用正则表达式完成对密码值的判断，并且根据不同的值定义输出不同的图像，其关键代码如下：

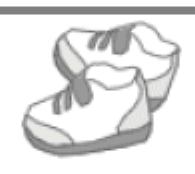
```
var p1 = (form.pwd.value.search(/[a-zA-Z]/)!=-1)?1:0;          //判断密码中是否包含英文字符串
var p2 = (form.pwd.value.search(/[0-9]/)!=-1)?1:0;             //判断密码中是否包含数字
var p3 = (form.pwd.value.search(/[^a-zA-Z0-9_]/)!=-1)?1:0;     //判断密码中是否包含特殊字符
var p = p1+p2+p3;                                              //定义判断的返回值
if(p==1){
        chknew_pwd.innerHTML="<img src='images/1_03.jpg'>";  //输出密码安全强度差的图像
    }else if(p==2){
        chknew_pwd.innerHTML="<img src='images/2_03.jpg'>";  //输出密码安全强度中的图像
    }else if(p==3){
        chknew_pwd.innerHTML="<img src='images/3_03.jpg'>";  //输出密码安全强度强的图像
    }
```

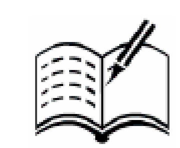
应用示例

在本示例中实现一个用户注册功能，通过 JavaScript 脚本判断密码值的强度，并且以图像的形式进行输出。其具体步骤如下：

（1）创建 index.php 页面，完成用户注册页面和表单的设计，通过 JavaScript 对表单中提交的值进行判断，并直接将判断的结果反馈到 div 标签中，最终将表单中的数据提交到 index_ok.php 文件中，完成用户注册的操作。其关键代码如下：

```
<form name="form_reg" method="post" action="index_ok.php" onSubmit="return chkreginfo
(form_reg,'all')">
<tr>
        <td height="30"><div align="right">密码：</div></td>
        <td height="30" colspan="2" align="left"> 
<input type="password" name="pwd" id="pwd" size="30" class="inputcss"
                onBlur="chkreginfo(form_reg,1)"> <font color="#FF0000">*</font> 
            <div id="chknew_pwd" style="color: #FF0000"></div>
        </td>
    </tr>
<tr>
        <td height="30"> </td>
        <td width="150" height="30"><input type="image" src="images/form (2).jpg">  
 </td>
        <td width="343"><img src="images/form.jpg" width="72" height="26"
                onClick="form_reg.reset()" style="cursor: hand" /></td>
    </tr>
</form>
```





（2）编写 JavaScript 脚本，创建 check.js 脚本文件，完成对表单中提交的值的判断。

（3）创建 index_ok.php 文件，获取表单中提交的数据，将数据添加到指定的数据表中。

运行本示例，当在密码文本框中输入一个密码时，光标离开后，将自动判断这个密码的安全强度，并以图像的形式返回一个判断结果。其运行结果如图 6.7 所示。



图 6.7　通过图像显示密码安全强度

专家点评

在本实例中，不止一次使用了正则表达式，分别通过正则表达式对密码的强度进行判断，对手机号码进行判断，对座机号码进行判断以及对邮箱地址进行判断。

```
/^13(\d{9})$|^15(\d{9})$|^189(\d{8})$/              //判断手机号码
/^(\d{3}-)(\d{8})$|^(\d{4}-)(\d{7})$|^(\d{4}-)(\d{8})$/   //判断座机号码
/\w+([-+.']\w+)*@\w+([-.]\w+)*\.\w+([-.]\w+)*/        //判断邮箱
```

问题 148　如何从图像中提取 EXIF 信息？

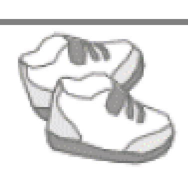
问题阐述

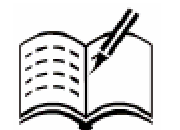
EXIF 信息就是由数码相机在拍摄过程中采集的一系列信息，然后把信息放置在 JPEG/TIFF 文件的头部，也就是说，EXIF 信息是镶嵌在 JPEG/TIFF 图像文件格式内的一组拍摄参数，主要包括摄影时的光圈、快门、ISO、日期时间等各种与当时摄影条件相关的信息，相机品牌型号，色彩编码，拍摄时录制的声音以及全球定位系统（GPS）等信息。既然在 EXIF 中包含了如此详细和完整的信息，那么在 PHP 中应该如何获取到这些信息呢？

专家解答

在 PHP 中获取这些信息应用的是 EXIF 函数。为了使用 EXIF 函数，必须保证 PHP 已经启用了 EXIF 支持。在 php.ini 文件中查看“extension=php_exif.dll”前是否有分号“;”，如果有则说明未加载，那么就要去掉分号。

同时还要注意“extension=php_exif.dll”项的存储位置，它必须放置在“extension=php_mbstring.dll”项之后，否则 EXIF 函数是不被支持的。设置完成后保存 php.ini 文件，并重新启





Note

动 Apache 服务器。然后就可以应用 EXIF 函数库中的函数获取图像的 EXIF 信息。

获取图像的 EXIF 信息应用的是 EXIF 函数库中的 exif_read_data()函数。其语法如下：

```
array exif_read_data ( string filename [, string sections [, bool arrays [, bool thumbnail]]] )
```

参数说明：

❶ filename：被读取的图像文件名。不可以是 URL。

❷ sections：定义存在于文件中的逗号分隔的区段列表用来产生结果数组。如果未找到所请求的区段则返回值为 False。该参数所包含的区段如表 6.2 所示。

表 6.2　exif_read_data()函数中参数 sections 包含的区段

区段名称	包含内容、说明
FILE	FileName、FileSize、FileDateTime、SectionsFound
COMPUTED	html、Width、Height、IsColor，可能有更多其他的。Height和Width是用和getimagesize()一样的方法计算的，因此它们的值不能是任何返回的头信息的部分。此外，html是一个height/width的文本字符串，可以用于普通的HTML中
ANY_TAG	任何包含有标记的信息，如IFD0、EXIF...
IFD0	所有 IFD0 的标记数据。在标准的图像文件中包含了图像大小及其他
THUMBNAIL	如果有第二个IFD，则文件应包含有缩略图。所有有关嵌入缩略图的标记信息都存储在本区
COMMENT	JPEG 图像的注释头信息
EXIF	EXIF 区段是 IFDO 的子区，包含有图像的更多详细信息。大多数内容都是与数码相机相关的

❸ arrays：指定是否每个区段都成为一个数组。sections COMPUTED、THUMBNAIL 和 COMMENT 区段总是成为数组，因为它们里面包含的名字和其他区段冲突。

❹ thumbnail：如果该值为 True，读取缩略图本身；否则只读取标记数据。

该函数的返回值是一个关联数组，键名是头信息名，值是与其对应的数据。如果没有可供返回的数据，则 exif_read_data()将返回 False。

应用示例

下面应用 exif_read_data()函数获取图像的 EXIF 信息。在分页输出上传图片时，运行自定义函数 GetImageInfo 获取图片的 EXIF 信息，并且输出图片的名称和拍摄时间。

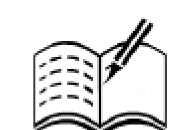
（1）在 index.php 页面中，实现上传图片的分页输出，并且包含 exif.php 文件，调用自定义函数 GetImageInfo 获取图片的 EXIF 信息，输出图片的名称和拍摄时间。其关键代码如下：

```
    <?php
    include "exif.php";                    //包含 exif.php 文件
$pic=GetImageInfo($myrow[file_test]);      //调用 getimageinfo 方法，获取原始图片的 exif 信息
echo "文件名：".$pic['文件名']."<br>";      //输出图片名称
echo " 拍摄时间：".$pic['拍摄时间'];        //输出拍摄时间
?>
```

（2）创建 exif.php 文件。编写自定义函数 getimageinfo()，应用 exif_read_data()函数获取指定图片的 exif 信息。其关键代码如下：

```
<?php
function GetImageInfo($img) {
```





```
                $imgtype = array ("", "GIF", "JPG", "PNG", "SWF", "PSD", "BMP" );
                $Orientation = array ("", "top left side", "top right side", "bottom right side" );
                $ResolutionUnit = array ("", "", "英寸", "厘米" );
                $exif = exif_read_data ( $img, "IFD0" );
                if ($exif === false) {
                    $new_img_info = array ("文件信息" => "没有图片 EXIF 信息" );
                } else {
                    $exif = exif_read_data ( $img, 0, true );
                    $new_img_info = array ("文件名" => $exif [FILE] [FileName], "时间戳" => date
( "Y-m-d H:i:s", $exif [FILE] [FileDateTime] );
                }
                return $new_img_info;
        }
        ?>
```

问题 149　如何应用 JQuery 和 Jpgraph 制作动态折线图？

问题阐述

如何应用 JQuery 和 Jpgraph 制作动态折线图？

专家解答

应用 JQuery 和 Jpgraph 制作动态折线图主要是应用 JQuery 中的属性实现数据的动态获取，应用 Jpgraph 生成折线图。

1. JQuery 中的属性函数

attr(key,fn)为所有匹配的元素设置一个计算的属性值。其不提供值，而是提供一个函数，由这个函数计算的值作为属性值。

参数 key(String)表示属性名称；参数 fn (Function)表示返回值的函数。其范围是当前元素，参数是当前元素的索引值。

示例：把 src 属性的值设置为 title 属性的值。

```
<img src="test.jpg"/>
$("img").attr("title", function() { return this.src });
```

输出结果:

```
<img src="test.jpg" title="test.jpg" />
```

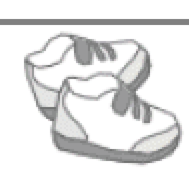
attr(key,value)，为所有匹配的元素设置一个属性值。

参数 key (String)是属性名称；参数 value (Object)是属性值。

示例：为指定图像设置 src 属性。

```
<img id="img" src="img.php?m=0" />
function beginProgress(counts){                          //创建自定义函数
  $("#img").attr("src", "img.php?m="+counts);            //为图像设置 src 属性
}
```

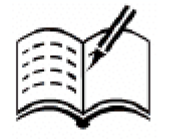
attr(name)，取得第一个匹配元素的属性值。通过这个方法可以方便地从第一个匹配元素中



获取一个属性的值。如果元素没有相应属性，则返回 undefined。

参数 name (String)是属性名称。

示例：返回页面中第一个图像的 src 属性值。





```
<img src="test.jpg"/>
$("img").attr("src");
```

输出结果：test.jpg

attr(properties)，将一个“名/值”形式的对象设置为所有匹配元素的属性。这是一种在所有匹配元素中批量设置很多属性的最佳方式。

参数 properties (Map)是作为属性的“名/值对”对象。

示例：为所有图像设置 src 和 alt 属性。

```
$("img").attr({ src: "test.jpg", alt: "Test Image" });
```

说明：

如果要设置对象的 class 属性，则必须使用 className 作为属性名，或者可以直接使用.addClass(class)和.removeClass(class)。

removeAttr(name)，从每一个匹配的元素中删除一个属性。

参数 name (String)是要删除的属性名。

示例：将页面中图像的 src 属性删除。

```
<img src="test.jpg"/>
$("img").removeAttr("src");
```

2. Jpgraph 下载与安装

要想应用 Jpgraph，首先要下载并安装 Jpgraph，同时下载该类库的压缩包。Jpgraph 类库的压缩包主要有两种形式，即 zip 格式和 tar 格式，如果是 Linux/UNIX 平台可以选择 tar 格式的压缩包；如果是微软的 Win32 平台选择上述两种格式的压缩包任一种都可以。Jpgraph 类库可以从其官方网站“http://www.aditus.nu/jpgraph/”下载，目前最新的版本是 2.3。

Jpgraph 需要 GD 库的支持。如果用户希望 Jpgraph 类库仅对当前站点有效，只需将 Jpgraph 压缩包下的 src 文件夹中的全部文件复制到网站所在目录下即可，在使用时，只需调用 src 文件夹下的指定文件即可。

说明：

如果想要让整个 PHP 开发环境支持 Jpgraph，那么可以使用下面的配置方法。(1）将压缩包下的全部文件解压到一个文件夹中，如“F:\AppServ\www\jpgraph”。(2）打开 PHP 的安装目录，编辑 php.ini 文件并修改其中的 include_path 参数，在其后增加前面的文件夹名，如 include_path = ".;F:\AppServ\www\jpgraph"。(3）重新启动 Apache 服务器即可生效

3. 应用示例

本示例应用 jQuery 和 Jpgraph，开发一个能够随时更新的动态折线图，通过该折线图分析网站当月访问量的走势。其运行结果如图 6.8 所示。





图 6.8　应用 JQuery 和 Jpgraph 制作动态折线图

（1）创建 index.php 文件，以数字图片的形式输出网站总访问量，并创建“网站访问量分析”的超链接。其关键代码如下：

```
<td height="50" align="center">
<?php
    //以图形的形式输出数据库中的记录数
        $query="select sum(counts) as ll from tb_count10 ";     //查询数据库中总的访问量
        $result=mysql_query($query);
        $fwl=mysql_result($result,0,'ll');
    echo "----------";
    //对补位数字 0 的处理
    $len=strlen($fwl);                                          //获取字符串的长度
    $str=str_repeat("0",6-$len);                                //获取 6-$len 个数字 0
    for($i=0;$i<strlen($str);$i++){                             //获取变量$str 的字符串长度
        $result=$str[$i];
        $result='<img src=images/0.gif>';
        echo $result;                                           //循环输出$result 的结果
    }
    //对数据库中数据的处理
    for($i=0;$i<strlen($fwl);$i++){                             //获取字符串的长度
        $result=$fwl[$i];
            switch($result){
            //如果值为"0",则输出 0.gif 图片
                case "0"; $ret[$i]="0.gif";break;
                case "1"; $ret[$i]="1.gif";break;
                case "2"; $ret[$i]="2.gif";break;
                case "3"; $ret[$i]="3.gif";break;
                case "4"; $ret[$i]="4.gif";break;
                case "5"; $ret[$i]="5.gif";break;
                case "6"; $ret[$i]="6.gif";break;
                case "7"; $ret[$i]="7.gif";break;
                case "8"; $ret[$i]="8.gif";break;
                case "9"; $ret[$i]="9.gif";break;
        }
```



```
            echo "<img src=images/".$ret[$i].".>";                  //输出访问次数
        }
    ?>
        <a href="search.php" class="STYLE1">网站访问量统计分析</a>
    </td>
```



Note

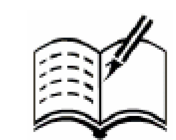
（2）创建 search.php 文件。对网站总的访问量、总的 IP 访问量，当月访问量、当月 IP 访问量，当日访问量、当日 IP 访问量进行统计，并且输出统计结果。其关键代码如下：

```
<td height="25" align="center" valign="middle"><?php
    $query_1="select sum(counts) as countes from tb_count10 ";
    //查询数据库中总的访问量
    $result_1=mysql_query($query_1);
    $countes_1=mysql_result($result_1,0,'countes');
    echo "<p class='STYLE2'>网站总的访问量：$countes_1</p>";
    //查询数据库中总的 IP 访问量
    $query_2="select * from tb_count10 ";
    $result_2=mysql_query($query_2);
    while($myrow=mysql_fetch_array($result_2)){
        $counts_2[]=$myrow[ip];                     //将获取的 IP 的值赋给变量
    }
    $results_2=array_unique($counts_2);             //去除数组中重复的值
    $countes_2=count($results_2);                   //获取数组中值的数量，即总的 IP 访问量
    echo "<p class='STYLE2'>网站总的 IP 访问量：$countes_2</p>";
?></td>
                <td width="323" align="center" valign="middle"><?php
    //查询数据库中当月总的访问量
    $query_3="select sum(counts) as countes from tb_count10 where data2='".substr(date("Y-m-d"),0,7)."'";
    $result_3=mysql_query($query_3);
    $countes_3=mysql_result($result_3,0,'countes');
?>
    <p class='STYLE2'>网站当月的访问量：<?php echo $countes_3;?></p>
    <?php

    //查询数据库中当月的 IP 访问量
    $query_4="select * from tb_count10 where data2='".substr(date("Y-m-d"),0,7)."'";
    $result_4=mysql_query($query_4);
    while($myrow_4=mysql_fetch_array($result_4)){
        $counts_4[]=$myrow_4[ip];                   //将获取的 IP 的值赋给变量
    }
    $results_4=array_unique($counts_4);             //去除数组中重复的值
    $countes_4=count($results_4);                   //获取数组中值的数量，即总的 IP 访问量
    echo "<p class='STYLE2'>网站当月 IP 的访问量：$countes_4</p>";
?></td>
                <td width="294" align="center" valign="middle"><?php
    //查询数据库中当日总的访问量
    $query_5="select sum(counts) as countes from tb_count10 where data1='".date("Y-m-d")."' ";
    $result_5=mysql_query($query_5);
    $countes_5=mysql_result($result_5,0,'countes');
    echo "<p class='STYLE2'>网站当日的访问量：$countes_5</p>";
    //查询数据库中当日的 IP 访问量
    $query_6="select * from tb_count10 where data1='".date("Y-m-d")."'";
```



```
            $result_6=mysql_query($query_6);
            while($myrow_6=mysql_fetch_array($result_6)){
                $counts_6[]=$myrow_6[ip];                    //将获取的 IP 的值赋给变量
            }
        if(is_array($counts_6)){
            $results_6=array_unique($counts_6);              //去除数组中重复的值
            $countes_6=count($results_6);                    //获取数组中值的数量，即总的 IP 访问量
            echo "<p class='STYLE2'>网站当日 IP 的访问量：$countes_6</p>";
        }
        ?></td>
        </tr>
```



接着，创建 form 表单，通过下拉列表将指定月份的数据提交到 progress()函数中，完成对指定月份网站访问量的分析。其关键代码如下：

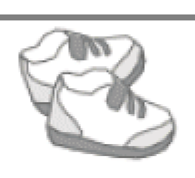
```
<form name="form1" method="post">
    <select name="selects" id="selects">
        <option value="2008-01">2008-01</option>
        <option value="2008-02">2008-02</option>
        <option value="2008-03">2008-03</option>
        <option value="2008-04">2008-04</option>
    </select>  
    <input type="button" name="Submit" value="网站月访问量分析" onClick="progress()">
</form>
```

最后，创建 img 图像，src 属性指定 img.php 文件，该文件的作用是通过 Jpgraph 生成网站月访问量的折线分析图。

```
<td colspan="2" align="center" bgcolor="#FFFFFF"><img id="img" src="img.php?m=0" /></td>
```

（3）创建 img.php 文件，通过 Jpgraph 生成网站月访问量的折线分析图。这里根据超链接中传递的参数 m 的值，执行查询操作获取数据库中的数据，获取到的数据就是折线图中的数据。其关键代码如下：

```
<?php
require_once 'src/jpgraph.php';                          //导入 Jpgraph 类库
require_once 'src/jpgraph_line.php';                     //导入 Jpgraph 类库的柱状图功能
include("conn/conn.php");                                //连接数据库
$data=$_GET['m'];                                        //获取超链接传递的参数
$query="select counts from tb_count10 where data2='$data' order by data1";
$result=mysql_query($query);                             //执行查询操作
while($myrow=mysql_fetch_array($result)){
    $dataTmp[]=current($myrow);                          //将数组中的当前单元添加到数组中
}
$data1 = array(0, 0);       //设置统计数据
for($i=0; $i<count($dataTmp); $i++){                     //执行 for 循环
    $data1[$i] = $dataTmp[$i];                           //获取数组中的数据
}
$graph = new Graph(800, 350);                            //定义画布大小
$graph->SetScale("textlin");                             //设置刻度样式
$graph->SetShadow();                                     //设置背景带阴影
$graph->img->SetMargin(50, 40, 40, 70);                  //设置统计图边距范围
$graph->title->Set("$data--网站流量分析");                //设置统计图标题
```






```
$lineplot1 = new LinePlot($data1);                    //建立 LinePlot 对象
$graph->Add($lineplot1);
$graph->xaxis->title->Set("天");                      //设置横坐标名称
$graph->yaxis->title->Set("网站流量分析（个）");      //设置第左侧纵坐标名称
$graph->title->SetFont(FF_SIMSUN, FS_BOLD);           //设置标题字体
$graph->yaxis->title->SetFont(FF_SIMSUN, FS_BOLD);    //设置 X 轴字体
$graph->xaxis->title->SetFont(FF_SIMSUN, FS_BOLD);    //设置 Y 轴字体
$lineplot1->SetColor('red');                          //设置折线颜色
$lineplot1->SetLegend($data."--"."网站访问流量");
$graph->legend->SetFont(FF_SIMSUN, FS_BOLD);
$graph->legend->SetLayout(LEGEND_HOR);
$graph->legend->Pos(0.4, 0.96, 'center', 'bottom');
$graph->Stroke();
?>
```

（4）在根目录下创建 js 文件夹，在 js 文件夹下创建 fun.js 文件，并将 JQuery 的文件包复制到 js 文件夹下。

在 fun.js 文件中编写自定义函数，通过 JQuery 属性中的函数实现对折线图中数据的动态加载。其中，progress()函数用于获取指定月份的数据，从而执行 beginProgress()函数，完成对指定月份网站流量的分析；而 progres()函数用于获取当前月份的数据，同时也执行 beginProgress()函数，对当前月份的网站访问量进行分析；在 beginProgress()函数中应用 JQuery 中的 attr 属性，设置图像 src 的属性，实现折线图的动态更新。其代码如下：

```
function progress(){                       //创建自定义函数
    counts=form1.selects.value;            //获取表单下拉列表中的值
        beginProgress(counts);             //将获取的值作为参数，执行自定义函数 beginProgress()
}
function progres(){                        //创建自定义函数
    var date_time=new Date();              //创建一个 Date 对象
    with(date_time){
    //定义变量，并为其赋值为当前年份，后加中文“年”字标识
    var date_times=getYear()+"-";
    //获取当前月份。注意月份从 0 开始，所以需加 1，后加中文“月”标识
    date_times+=getMonth()+1;              //在 JavaScript 脚本中获取当前时间
    }
    beginProgress(date_times);             //将获取的时间作为参数，执行自定义函数 beginProgress
    setTimeout("progres()", 60000);        //间隔一段时间自动执行一次 progres 函数
}
function beginProgress(counts){                       //创建自定义函数
    $("#img").attr("src", "img.php?m="+counts);       //为图像设置 src 属性
}
```

问题 150　如何通过 Jpgraph 类库输出中文字符串？

问题阐述

Jpgraph 类库是一个可以应用在 PHP4.3.1 以上版本的用于图形图像绘制的类库，该类库完全基于 GD2 函数库编写，提供了多种方法创建各类统计图，包括坐标图、柱状图、饼形图等。



由于 Jpgraph 类库在编写时采用的是 UTF-8 编码，所以它在输出中文字符串时会出现乱码的问题，对此虽然网络上有很多的解决方案，但是效果一般。因此，我们在这里整理出一个有效的解决方案，供读者参考和学习。

专家解答

解决 Jpgraph 类库中输出中文字符串乱码的问题，可以分 3 步走：

第一步：创建编码格式为 UTF-8 的网页文件。

第二步：通过 iconv()函数实现中文字符串编码格式的转换，将 UTF-8 编码转换为 gb2312 编码。

第三步：设置中文字符串的字体（简体为 FF_SIMSUN, FS_BOLD，繁体为 FF_CHINESE）。

通过上述 3 步即可圆满地解决 Jpgraph 类库中输出中文字符串乱码的问题。

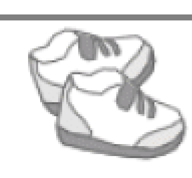
注意：

此处应用的 Jpgraph 的版本是 3.0.7。

应用示例

应用 Jpgraph 类库创建一个 3D 饼形图，输出 MySQL 图书各部分内容所占比例。创建一个 UTF-8 编码格式的 index.php 文件，载入 Jpgraph 类库文件 jpgraph.php、jpgraph_pie.php 和 jpgraph_pie3d.php，调用类库中的方法，完成 3D 饼形图的设计。其关键代码如下：

```
<?php
header ( "Content-type: text/html; charset=UTF-8" );          //设置文件编码格式
include ("../src/jpgraph.php");
include ("../src/jpgraph_pie.php");
include ("../src/jpgraph_pie3d.php");
$data = array(8,5,4,3);                                       //定义数据大小
$graph = new PieGraph(750,450,"auto");                        //创建图像
$graph->SetShadow();                                          //创建图像阴影
$graph->tabtitle->Set(iconv("utf-8","gb2312",'MySQL 图书内容分布比例') );
                                                              //输出标题，进行编码格式转换
$graph->tabtitle->SetFont(FF_SIMSUN, FS_BOLD,14);             //设置标题字体
$graph->title->SetColor("darkblue");                          //定义标题颜色
$graph->legend->SetFont(FF_SIMSUN, FS_BOLD,14);               //设置辅助标题字体
$graph->legend->Pos(0.03,0.1);                                //控制注释文字的位置
$p1 = new PiePlot3d($data);                                   //创建图像
$p1->SetTheme("sand");                                        //控制图像颜色
$p1->SetCenter(0.35);                                         //设置图像位置
$p1->SetSize(0.4);                                            //设置图像大小
$p1->SetHeight(40);                                           //设置饼形图高度
$p1->SetAngle(45);                                            //设置图像倾斜角度
$p1->Explode(array(32,20,16,18));                             //控制饼形图的分割
$p1->value->SetFont(FF_ARIAL,FS_NORMAL,10);                   //设置内容字体和字号大小
$p1->SetLegends(array(iconv("utf-8","gb2312","MySQL 基础篇"),iconv("utf-8","gb2312", "MySQL 高级应用篇"),iconv("utf-8","gb2312","MySQL 系统管理篇"),iconv("utf-8","gb2312", "MySQL 应用程序开发篇")));
                                                              //输出辅助标题内容
$graph->Add($p1);                                             //添加数据
$graph->Stroke();                                             //生成图像
?>
```





Note

运行结果如图 6.9 所示。

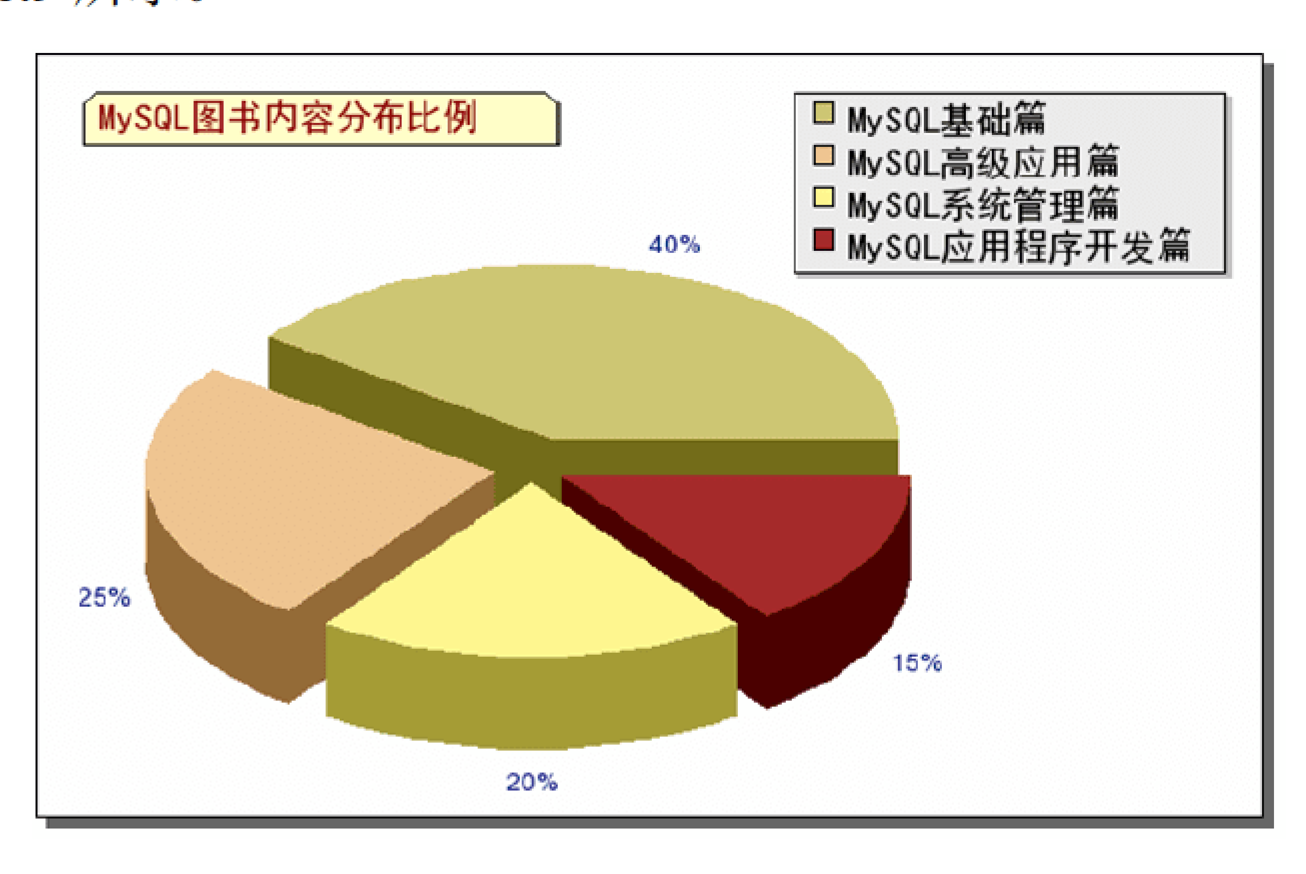


图 6.9　输出中文字符串

专家点评

通过此方法在 Jpgraph 类库中输出中文适用于 Jpgraph 类库的各系列版本。另外，如果用户通过 XAMPP 集成化安装包配置的 PHP 开发环境，那么在应用此方法输出中文时有可能也会出现乱码，因为它在处理 GD 库方面存在一些问题，至于使用其他的集成化安装包或手动配置的 PHP 开发环境，使用上述方法都可以完成中文的正常输出。

问题 151　如何制作数字图像验证码？

问题阐述

为了防止用户通过恶意程序登录站点，提高网站的安全性，项目开发时，在用户登录、注册及发表主题等模块中加入验证码模块是必不可少的。应用验证码技术可以有效地防止用户通过探测的方式非法登录，从而有效地提高站点的安全性。

专家解答

制作数字图像验证码，首先，通过 JavaScript 中 Math 对象的 random()方法随机生成 4 位随机数，并通过 doument 对象的 write()方法将这 4 位随机数所对应的图片以链接形式输出。

然后，通过$_GET[]全局数组提取链接地址所传入的随机数值，通过 PHP 文件操作函数读取该随机数对应的图片，并以二进制的形式输出到浏览器。

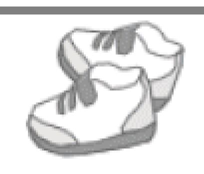
接着，定义一个重新指定图片链接地址的方法，当单击“看不清”超链接时，调用该方法生成新的验证码图片。

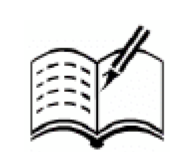
最后，在表单中添加一个隐藏域，用于提交 4 位随机数，当用户单击表单中的“登录”按钮时，通过 JavaScript 将隐藏域的值与验证码文本框所输入的内容进行比较，完成验证操作。

（1）创建 4 位随机数

通过 JavaScript 中 Math 对象的 random()方法随机生成 4 位随机数。关键代码如下：

```
var num1=Math.round(Math.random()*10000000);
var num=num1.toString().substr(0,4);                    //生成 4 位随机数
```





（2）将图片以二进制形式输出

应用 PHP 的文件操作函数即可实现将图片以二进制的形式输出到浏览器，应用 fread()函数实现，其语法如下：

```
fread ( int handle, int length)
```

参数说明：

❶ handle：fopen()函数打开文件后所返回的文件句柄。

❷ length：所读取文件的长度。

例如，本实例中应用 fread()方法将图片以二进制形式输出到浏览器，实现代码如下：

```
$fp=fopen($address,"r");                        //打开文件
echo fread($fp,filesize($address));             //读取文件
```

应用示例

在本示例中，应用 JavaScript 技术，再以 0 到 9 这 10 个别致数字图片为辅助，制作博客后台登录的验证码。其具体步骤如下：

（1）编写 index.php 用户登录页面，创建表单，在表单中添加用户名文本域、密码域、验证码文本域以及保存随机数的隐藏域。

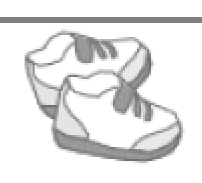
（2）创建 fun.js 脚本文件，在客户端对表单提交数据进行验证。

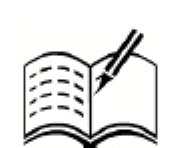
（3）编写 JavaScript 脚本生成 4 位随机数，并且在用户单击“看不清”超链接时，调用自定义方法 code_1()动态更改验证码的值。其关键代码如下：

```
<script language="javascript">
var num1=Math.round(Math.random()*10000000);
    var num=num1.toString().substr(0,4);            //产生 4 位随机数
    document.write("<img name=codeimg4 src='xym.php?num="+num.substr(0,1)+"'>");
    document.write("<img name=codeimg5 src='xym.php?num="+num.substr(1,1)+"'>");
    document.write("<img name=codeimg6 src='xym.php?num="+num.substr(2,1)+"'>");
    document.write("<img name=codeimg7 src='xym.php?num="+num.substr(3,1)+"'>");
    form_login.xym1.value=num;
    function code_1(){                              //通过该方法动态改变验证码的值
    var num1=Math.round(Math.random()*10000000);
        var num=num1.toString().substr(0,4);
        document.codeimg4.src="xym.php?num="+num.substr(0,1);
        document.codeimg5.src="xym.php?num="+num.substr(1,1);
        document.codeimg6.src="xym.php?num="+num.substr(2,1);
        document.codeimg7.src="xym.php?num="+num.substr(3,1);
        form_login.xym1.value=num;                  //将验证码的值赋给隐藏域
    }
</script>
```

（4）创建 xym.php 文件，采用 PHP 文件操作函数将图片以二进制的形式输出到浏览器，这样可以有效地提高静态页面与动态页面之间的交互性。其代码如下：

```
$num=$_GET[num];                                //接收 Get 方法传递过来的参数
$address="images/code/".$num.".gif";            //获取图片路径
$fp=fopen($address,"r");                        //打开图片文件
echo fread($fp,filesize($address));             //将图片文件读取到浏览器
fclose($fp);                                    //关闭打开的文件句柄
```





Note

运行本示例，其运行结果如图 6.10 所示。



图 6.10　数字图像验证码

专家点评

如果在 PHP 页面中不仅包含 PHP 代码，而且包含其他语言代码，如 JavaScript、CSS 或 HTML 等，则需要写完整的 PHP 代码块标记，即“<?php”与“?>”都要写；如果 PHP 页面代码完全由 PHP 构成，则建议只写开始标记，不写结束标记，这样可以防止恶意用户通过非法注入等手段在页面代码后注入恶意代码。

问题 152　如何制作中文图像验证码？

问题阐述

验证码不但可以使用数字图像来完成，而且可以使用中文图像来完成。那么如何制作中文图像验证码呢？

专家解答

与开发数字验证码不同，开发中文图像验证码，首先应确定所使用的文字，并且将文字制作成图片，以数字进行命名。然后，将所使用的文字定义到数组中。最后，计算出数组中元素的长度值，并应用 rand()函数从这个长度值中随机取值，根据随机取值输出对应的中文图像，即验证码。

应用示例

在本示例中，制作博客后台登录页面，并采用中文汉字生成验证码。其具体步骤如下：

（1）编写 index.php 用户登录页面，创建表单，在表单中添加用户名文本域、密码域、验证码文本域以及保存随机数的隐藏域。

（2）在 index.php 页面中，编写 PHP 脚本，定义验证码所使用的中文字符串，并生成验证码，其关键代码如下：

```
<?php
$str = array ("大", "更", "创", "天", "科", "客", "博", "技", "立", "新" );   //定义中文字符串的数组
$word = count ( $str );                    //计算出数组中元素个数
for($i = 0; $i < 4; $i ++) {               //执行 for 循环，以数组中元素个数为范围随机取出 4 个值
```



```
        $num = rand ( 0, $word - 1 ); $word=$word*2-1                //随机取值
    //输出随机取出的 4 个值对应的中文图片
        $img = $img . "<img src=' images/checkcode/" . $num . ".gif' width='16' height='16'>";
                                                                     //显示随机图片
        $pic = $pic . $str [$num];                                   //将图片转换成数组中的文字
    }
    ?>
```



（4）创建 checkuser.php 文件，完成对用户提交的登录信息的验证。

博客后台登录页面的运行结果如图 6.11 所示。



图 6.11　中文图像验证码

专家点评

在本示例中，应用 count()函数计算数组中元素的数目。注意，count()函数只适用于计算数组中元素的个数，如果将其应用到其他类型的数据中，那么返回值只有一个单元。如果要计算字符串的长度，那么要使用 strlen()函数，切记 strlen()函数不可对数组元素进行操作，虽然不返回错误信息，但是返回的结果也是不准确的。

问题 153　如何实现验证码在当前页验证？

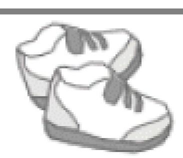
问题阐述

验证码技术是为了防止填写的信息被暴力破解，在签写信息时，通常被要求输入验证码。通常情况下，如果输入的验证码错误，将重新刷新页面，所签写的信息将被清空，这样给用户操作造成了一定的麻烦。为了避免这一现象的发生，就需实现让验证码在当前页验证，该如何实现？

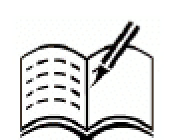
专家解答

实现验证码在当前页验证可以通过 JavaScript 脚本来实现，封装自定义函数 check_form()，获取表单提交的验证码，实现对验证码的验证。其代码如下：

```
<script language="javascript">
function check_form(form1){
    if(form1.checkcode.value==""){   //如果验证码为空
        alert("验证码不能为空！ ");form1.checkcode.focus();return false;
    }
```



```
            if(form1.checkcode.value!=num){     //如果输入的验证码的值与随机的验证码的值不相等
                  alert("您输入的验证码不正确，重新输入！");form1.checkcode.focus();return false;
            }
      }
```



Note

问题 154　如何任意调整图片的大小?

问题阐述

一般情况下，我们上传的图片都是需要手动修改统一大小，即使可以通过缩略图的方式统一控制上传图片的大小，但是有些时候不是很理想。那么是否有办法可以任意在程序中调整图片的大小呢?

专家解答

任意调整上传图片大小方法实现的原理：

首先，根据表单中提交的值获取指定图片的存储路径和名称以及它要被调整的比例。

然后，应用 getimagesize()函数获取指定图片的数据。

接着，定义 thumb()方法，根据原始图片的数据和提交的调整比例数据，判断是根据比例缩放还是进行裁剪。

最后，载入原始图片，并定义新的缩略图，应用 ImageCopyResampled()函数将原始图像复制到新的缩略图中，并将新的图像存储到指定位置。

imagecopyresampled()函数，从原图像中采样，复制部分图像，重新生成图像。

Imagecopyresized()函数，同样实现图像的复制操作。语法如下：

```
      bool imagecopyresized ( resource dst_image, resource src_image, int dst_x, int dst_y, int src_x, int
src_y, int dst_w, int dst_h, int src_w, int src_h )
```

imagecopyresized() 将一幅图像中的一块正方形区域复制到另一个图像中。

参数说明：

❶ dst_image：是目标图像标识符。

❷ src_image：是源图像标识符。

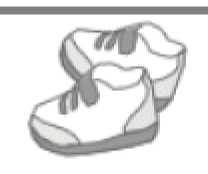
如果源图像和目标图像的宽度和高度不同，则会进行相应的图像收缩和拉伸。坐标指的是左上角。本函数可用来在同一幅图内部复制（如果 dst_image 和 src_image 相同）区域，但如果区域交迭则结果不可预知。

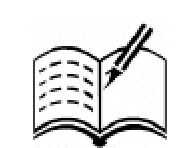
说明：

因为调色板图像限制（255+1 种颜色）有个问题。重新采样或过滤图像通常需要多于 255 种颜色，计算新的被重新采样的像素及其颜色时采用了一种近似值。对调色板图像尝试分配一个新颜色时，如果失败那么选择计算结果最接近（理论上）的颜色，这并不是视觉上最接近的颜色。这可能会产生怪异的结果，如空白（或者视觉上是空白）的图像。要跳过这个问题，应使用真彩色图像作为目标图像，如使用 imagecreatetruecolor()函数创建的图像。

应用示例

在本示例中，开发一个图片上传的功能，并且可以对上传图片大小任意调整。其具体步骤





如下：

（1）创建一个图片上传功能，将图片上传到服务器指定的文件夹下，并通过文件系统函数读取出存储在指定目录下的图片。

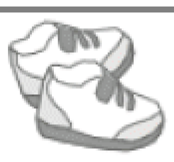
（2）在图片输出的同时，在右侧创建一个表单，提交图片的存储位置、名称和要调整的尺寸，将数据提交到 index_ok.php 文件中。

（3）创建 index_ok.php 文件，包含 thumbnail.php 文件，实例化 image 类，调用其中的 thumb()方法，将表单中提交的数据作为参数，完成对图片大小的调整。其关键代码如下：

```
<?php
if($_POST['Submit']=="调整图像大小"){
include("thumbnail.php");                                    //包含图像类存储文件
$img = new image();                                          //实例化图像处理类
$return=$img->param($_POST['pic_path'])->thumb("./thumbnail/".$_POST['pic_name'], $_POST['pic_width'],
$_POST['pic_height'],1);
echo "<script>alert('图像大小设置成功！'); window.location.href='index.php';</script>";
}
?>
```

（4）创建 thumbnail.php 文件，编写 image 类，定义 thumb()方法，根据表单中提交的数据，完成图像大小的调整操作。其关键代码如下：

```
//thumb(新图地址, 宽, 高, 裁剪, 允许放大)
public function thumb($filename, $news_w = 100, $news_h = 100, $cut = 0, $big = 0) {
    $info = $this->getImageInfo ( $this->img );              //获取原图信息
    if (! empty ( $info [0] )) {
        $old_w = $info [0];
        $old_h = $info [1];
        $new_w = $old_w * $news_w;
        $new_h = $old_h * $news_h;
        $type = $info ['type'];
        $ext = $info ['ext'];
        unset ( $info );
        if ($old_w < $new_h && $old_h < $new_w && ! $big) {    //如果原图比缩略图小
                                                               并且不允许放大
            return false;
        }
        if ($cut == 1) {                                       //居中裁剪
            $scale1 = round ( $new_w / $new_h, 2 );
            $scale2 = round ( $old_w / $old_h, 2 );
            if ($scale1 > $scale2) {
                $end_h = round ( $old_w / $scale1, 2 );
                $start_h = ($old_h - $end_h) / 2;
                $start_w = 0;
                $end_w = $old_w;
            } else {
                $end_w = round ( $old_h * $scale1, 2 );
                $start_w = ($old_w - $end_w) / 2;
                $start_h = 0;
                $end_h = $old_h;
            }
```




```
                $width = $new_w;                                    //定义图像宽
                $height = $new_h;                                   //定义图像高
            }
            $createFun = 'ImageCreateFrom' . $type;                 //新建图像
            $oldimg = $createFun ( $this->img );
            if ($type != 'gif' && function_exists ( 'imagecreatetruecolor' )) {
                $newimg = imagecreatetruecolor ( $width, $height );     //创建新的真彩图像
            } else {
                $newimg = imagecreate ( $width, $height );              //创建新的图像
            }
            if (function_exists ( "ImageCopyResampled" )) {
                ImageCopyResampled ( $newimg, $oldimg, 0, 0, $start_w, $start_h, $width,
$height, $end_w, $end_h );                                          //完成图像的复制
            } else {
                ImageCopyResized ( $newimg, $oldimg, 0, 0, $start_w, $start_h, $width, $height,
$end_w, $end_h );
            }
            $type == 'jpeg' && imageinterlace ( $newimg, 1 );       //对 jpeg 图形设置隔行扫描
            $imageFun = 'image' . $type;                            //生成图片
            ! @$imageFun ( $newimg, $filename ) && die ( '保存失败!' );
            ImageDestroy ( $newimg );                               //销毁图像
            return $filename;                                       //返回图像名称
        }
    }
```

运行本示例，在上传图片的右侧选择要调整图片大小的比例，然后单击下面的按钮，其运行结果如图 6.12 所示。



图 6.12　任意调整图像的大小



文件与目录操作



问题 155　如何限制上传文件的大小?



Note

问题阐述

在网站开发的过程中，为了确保能够充分利用服务器的空间，在开发上传功能时，必须对上传文件的大小进行控制。那么如何对上传文件的大小进行控制呢?

专家解答

控制上传文件的大小可以从两个方面入手：第一是在 PHP 的配置文件 php.ini 中对上传文件进行控制，如果上传文件超过它指定的范围，那么执行上传就会失败；第二是在 PHP 配置文件允许的范围内，在程序中对上传文件的大小进行控制。

1. 通过配置文件控制上传文件

PHP 中通过 php.ini 文件对上传文件进行控制，包括是否支持上传、上传文件的临时目录、上传文件的大小、指令执行的时间、指令分配的内存空间。

在 php.ini 中，定义到 File Uploads 项，完成对上传相关选项的设置。上传相关选项的含义如下：

- ☑ file_uploads：如果值是 on，说明服务器支持文件上传；如果为 off，则不支持。一般默认是支持的，这个不用修改。
- ☑ upload_tmp_dir：上传文件临时目录。在文件被成功上传之前，文件首先存放到服务器端的临时目录中。多数使用系统默认目录，但是也可以自行设置。
- ☑ upload_max_filesize：服务器允许上传文件的最大值，以 MB 为单位。系统默认为 2MB，如果网站需要上传超过 2MB 的数据，那么就要修改这个值。

上述是 php.ini 中 File_Uploads 项中与上传相关选项参数设置说明，除了 File_Uploads 项中的内容外，在 php.ini 中还有其他几个选项会影响到文件的上传。

- ☑ max_execution_time：PHP 中一个指令所能执行的最大时间，单位是秒。该选项在上传超大文件时必须要修改，否则即使上传文件在服务器允许的范围内，但是若超过了指令所能执行的最大时间，仍然无法实现上传。
- ☑ memory_limit：PHP 中一个指令所分配的内存空间，单位是 MB。它的大小同样会影响到超大文件的上传。

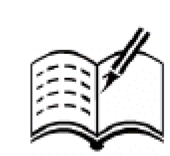
2. 在程序中控制上传文件

在客户端控制上传文件应用的是 form 表单中的 enctype 和 method 属性以及隐藏域 MAX_FILE_SIZE。

- ☑ enctype="multipart/form-data"：指定表单编码数据方式。
- ☑ method="post"：指定数据的传输方式。
- ☑ <input type="hidden" name="MAX_FILE_SIZE" value="10000" />：通过隐藏域控制上传文件的大小（单位为字节），该值不能超过 php.ini 配置文件中 upload_max_filesize 选项设置的值。它不能完全控制上传文件的大小，只是可以避免一些不必要的麻烦。







技巧：

在应用隐藏域<input type="hidden" name="MAX_FILE_SIZE" value="10000" />控制上传文件大小时，必须将其放置在文件域之前，否则是不能发挥作用的。

3. 通过全局变量$_FILES 获取上传信息

上述两种判断的结果都可以通过全局变量$_FILES 的返回值来体现。$_FILES 是一个数组，它包含所有上传的文件信息。下面介绍$_FILES 数组中每个元素的含义，如表 7.1 所示。

表 7.1　$_FILES 数组中的元素含义

元　素　名	说　　明
$_FILES[filename][name]	存储上传文件的文件名，如text.txt、title.jpg等
$_FILES[filename][size]	存储文件大小。单位为字节
$_FILES[filename][tmp_name]	存储文件在临时目录中使用的文件名。因为文件在上传时，首先要将其以临时文件的身份保存在临时目录中
$_FILES[filename][type]	存储上传文件的MIME类型，该类型规定各种文件格式的类型。每种MIME类型都是由“/”分隔的主类型和子类型组成的，如“image/gif”，主类型为“图像”，子类型为GIF格式的文件，“text/html”代表文本的HTML文件
$_FILES[filename][error]	存储与文件上传相关的错误代码。此项目是PHP 4.2.0 版本中新增内容。其返回值有5种，如下所示： 0：表示没有任何错误，文件上传成功 1：表示上传文件的大小超出了PHP配置文件指令upload_max_filesize选项限制的值 2：表示上传文件大小超出了HTML表单中MAX_FILE_SIZE选项所指定的值 3：表示文件只被上传了一部分 4：表示没有上传任何文件

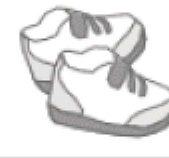
4. 应用示例

本示例实现一个文件上传功能，并且控制上传文件大小在 2MB 之内。如果超出大小，则返回相应的错误信息。其具体步骤如下：

（1）创建 index.php 文件。添加表单，设置隐藏域、文件域、提交按钮，使用 POST 方法，设置 enctype="multipart/form-data"，将数据提交到本页。

（2）通过$_FILES 获取上传文件的相关信息，通过 move_upload_file()函数完成图片上传。代码如下：

```
<?php
if(!empty($_FILES[up_picture][name])){              //判断上传内容是否为空
    if($_FILES['up_picture']['error']>0){           //判断文件是否可以上传到服务器
        echo "上传错误:";
        switch($_FILES['up_picture']['error']){
            case 1:
                echo "上传文件大小超出配置文件规定值";
            break;
```

```
                case 2:
                    echo "上传文件大小超出表单中约定值";
                break;
                case 3:
                    echo "上传文件不全";
                break;
                case 4:
                    echo "没有上传文件";
                break;
            }
        }else{
            if(!is_dir("./upfile/")){                                  //判断指定目录是否存在
                mkdir("./upfile/");                                    //创建目录
            }
            $path='./upfile/'.time().strstr($_FILES['up_picture']['name'],'.');//定义文件名称和存储位置
            if(is_uploaded_file($_FILES['up_picture']['tmp_name'])){      //是否是 HTTP POST 上传
                if(!move_uploaded_file($_FILES['up_picture']['tmp_name'],$path)){      //执行上传
                    echo "上传失败";
                }else{
                    echo "文件".time().$_FILES['up_picture']['name']."上传成功，大小为：".$_FILES
['up_picture']['size'];
                }
            }else{
                echo "上传文件".$_FILES['up_pictute']['name']."不合法！";
            }
        }
    }
    ?>
```

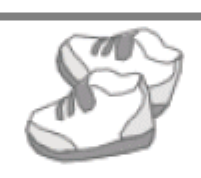
其运行结果如图 7.1 所示。



图 7.1　限制上传文件的大小

专家点评

如果要上传超大的文件，则必须修改 php.ini 文件。修改选项包括 upload_max_filesize 的最大值、max_execution_time 一个指令所能执行的最大时间和 memory_limit 一个指令所分配的内存空间；否则即使上传文件大小在规定的范围之内，由于指令的执行时间不足或者指令分配的内存不足都无法顺利地完成文件的上传操作。





问题 156　如何判断上传文件是否符合要求？

问题阐述

文件上传功能几乎是所有网站所必备的，然而文件上传对于服务器来说具有很大风险，因此对文件大小和文件类型进行限制就显得十分有必要。控制上传文件大小的问题在前面内容中已经解决，那么如何对上传文件的类型进行控制呢？

专家解答

限制上传文件类型，首先要通过预定义全局变量$_FILES 中的$_FILES['files']['name']获取上传文件的名称，然后应用 strstr()截取上传文件的后缀，最后判断上传文件的后缀是否符合要求。

strstr()函数的语法如下：

```
string strstr(string strings, string needle);
```

该函数用于在字符串 strings 中查找字符串 needle 的位置，并返回从 needle 字符串开始到 strings 字符串结束所有的内容，如果在 strings 中没有查找到 needle，则该函数将返回 false。

判断上传文件类型的方法如下：

```
<?php
if(!empty($_FILES['up_picture'][ 'name'])){            //判断上传内容是否为空
$type=$_FILES['up_picture']['name'];                   //获取上传文件的名称
$types=strstr($type,'.');                              //截取上传文件的后缀
if($types==".txt"){                                    //判断上传文件的后缀是否符合要求
        //省略了部分代码
    }
    }else{
            echo "上传文件".$_FILES['up_pictute']['name']."类型不正确！ ";
    }
}
?>
```

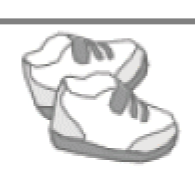
专家点评

判断上传文件是否符合多种类型中的一种最佳的方法就是将多种类型定义到一个数组中，然后通过 for 循环输出数组中的元素值，将输出的元素值与获取的文件后缀进行比较，进而判断其是否符合要求。

问题 157　如何通过 header()函数进行下载？

问题阐述

除了可以通过链接方式下载文件之外，还可以通过 header 函数进行下载，那么这个方法具体是如何实现的呢？



专家解答

通过 HTTP 方式下载文件，主要应用 header()函数。

1. header()函数





header()函数属于 HTTP 函数。其作用是以 HTTP 协议将 HTML 文档的标头送到浏览器，并告诉浏览器具体如何处理这个页面。header()函数的语法如下：

```
void header ( string string [, bool replace [, int http_response_code]] )
```

参数说明：

❶ string：发送的标头。

❷ replace：如果一次发送多个标头，对于相似的标头是替换还是添加。如果是 false，则强制发送多个同类型的标头。默认是 true，即替换。

❸ http_response_code：强制 HTTP 响应为指定值。

通过 HTTP 下载的代码如下：

```
header("Content-type: application/x-gzip");
header("Content-Disposition: attachment; filename=文件名");
header("Content-Description: PHP3 Generated Data"); >
```

HTTP 标头有很多，这里介绍的是下载的 HTTP 标头。其代码如下：

```
header('Content-Disposition: attachment; filename="filename"');
```

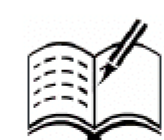
在应用的过程中，唯一需要改动的就是 filename，即将 filename 替换为要下载的文件。

2. 应用示例

根据超链接传递的参数值判断指定的文件是否存在，如果存在，则通过 header()函数完成文件的下载操作。其关键代码如下：

```
<?php
    header("Content-type: text/html; charset=UTF-8");     //设置文件编码格式
    $path = $_GET['path'];                                //获取文件路径
    if(!empty($path) and !is_null($path)){                //判断变量是否为空以及是否为 NULL
        $filename=basename($path);                        //获取文件名
        $file=@fopen($path,"r");
        if($file){
            header("Content-type:application/octet-stream");          //输出 MIME 类型
            header("Accept-ranges:bytes");                            //接受的范围单位
            header("Accept-length:".filesize($path));                 //文件长度
            header("Content-Disposition:attachment;filename=".$filename);
                                                          //默认时文件保存对话框中的文件名称
            echo fread($file,filesize($path));            //读取文件
            fclose($file);                                //关闭文件
            exit;                                         //退出
        }else{
            echo "<script>alert('您下载的文件不存在！ '); history.back();</script>";
        }
    }
?>
```





专家点评

通过 header()函数不但可以实现文件的下载，还可以实现下面 3 个功能：

（1）重定向，这是最常用的功能。

```
header("Location: http://www.mrbccd.com");
```

（2）强制客户端每次访问页面时获取最新资料，而不是使用存在于客户端的缓存。

```
//设置页面的过期时间(用格林威治时间表示)。
header("Expires:     Mon,     08     Jul     2008     08:08:08     GMT");
//设置页面的最后更新日期（用格林威治时间表示），使浏览器获取最新资料
header("Last-Modified: " . gmdate("D, d M Y H:i:s") . "GMT");
header("Cache-Control: no-cache,     must-revalidate");          //控制页面不使用缓存
header("Pragma: no-cache");              //参数（与以前的服务器兼容），即兼容 HTTP1.0 协议
header("Content-type:     application/file");                     //输出 MIME 类型
header("Content-Length:     227685");                             //文件长度
header("Accept-Ranges:     bytes");                               //接受的范围单位
//默认时文件保存对话框中的文件名称
header("Content－Disposition: attachment; filename=$filename");   //实现下载
```

（3）输出状态值到浏览器，控制访问权限。

```
header('HTTP/1.1 401 Unauthorized');
header('status: 401 Unauthorized');
```

问题 158　如何实现多文件同时上传？

问题阐述

上传图片到服务器是程序开发过程中必不可少的一项功能。使用该功能不但可以达到图片共享的效果，而且可以提高网站的访问量，丰富网站的内容。但是，在网站的后台管理系统中，需要的是更简单、更快速地上传图片到服务器，这就是多文件同时上传到服务器。那么这个功能是如何实现的呢？它的原理又是怎样的呢？

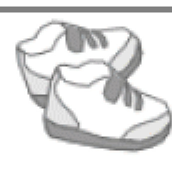
专家解答

多文件同时上传的关键是 for 循环语句的应用，通过 for 循环语句定义上传文件的文件域名称，由 for 循环语句决定一次最多可上传几个文件。

在处理提交的上传时，同样应用 for 循环语句，获取上传文件的数据，通过 move_uploaded_file()函数完成文件的上传操作。

1. 定义多文件上传表单

```
<form action="" method="post" enctype="multipart/form-data">
  <?php
      for($a=0;$a<4;$a++){
  ?>
          <input type="file" name="filename_<?php echo $a;?>">
  <?php
      }
```



```
    ?>
    <br><input class="one"type="submit"name="sub" value="上传"/>
    </form>
```



Note

应用 POST 方法设置 enctype 的属性为"multipart/form-data"，将数据提交到本页，应用 for 循环语句定义文件上传的文件域。

2. for 循环语句多文件上传处理

```
<?php
if(isset($_POST['sub'])){
        for($a=0;$a<4;$a++){                            //循环语句
            $name=$_FILES['filename_'.$a];              //将变量的名称保存在变量中
            $rand=rand(1,100000);                       //获取随机数
            $name_type=substr($name['name'],-4,4);      //获取文件名称的后 4 位即文件的格式
            $time=microtime();                          //获取随机数
            $path='upload/'.(($rand+$time).$name_type);               //定义上传文件的路径
            $dir='upload/';                                           //定义目录
            if(move_uploaded_file($name['tmp_name'],$path)){
                echo "成功上传文件".($rand+$time).$name_type."<br>"; //输出提示
            }                                           //移动到服务器指定位置
        }
    }
?>
```

应用$_FILES 全局数组以 for 循环语句为辅助，获取上传数据的信息，通过 rand()和 microtime()函数获取的数字作为文件名称，应用 substr()函数获取上传文件后缀，通过 move_uploaded_file()函数执行文件上传操作。

专家点评

这里讲解的多文件上传的方法，其文件上传的最大值取决于 for 循环语句设置的最大值，它一次可以上传最大值之内任意多个文件。

问题 159　如何判断文件的权限？

问题阐述

文件权限的判断是一个非常重要的功能，如果文件没有读、写的权限，那么就不能对其进行操作，那么如何判断文件的权限呢？

专家解答

判断文件是否具备读权限应用的是 is_readable()函数；判断文件是否具备写权限应用的是 is_writable()函数。

is_readable()函数，判断指定文件是否可读，语法如下：

```
bool is_readable ( string filename )
```

如果文件存在并且可读则返回 TRUE。



is_writable()函数，判断指定文件是否可写，语法如下：

```
bool is_writable ( string filename )
```

如果文件存在并且可写则返回 TRUE。filename 参数可以是一个允许进行是否可写检查的目录名。



说明：

PHP 也许只能以运行 webserver 的用户名（通常为'nobody'）来访问文件。不计入安全模式的限制。

对于文件权限的判断是对文件进行操作的前提，特别是在执行文件的读取、写入、重命名等操作时，如果文件不具备读、写的权限，那么这些操作都是没有任何意义的。

专家点评

通过 is_readable()和 is_writable()函数可以判断文件是否具有读、写的权限。在文件系统函数库中还可以通过 fileperms()函数取得文件的权限，它的返回值是一个 int 型的字符串。

问题 160　如何获取磁盘分区的大小？

问题阐述

通过文件系统函数不但可以对目录、文件进行操作，而且可以获取目录、文件的相关信息，那么是否可以获取磁盘分区的大小呢？

专家解答

获取磁盘分区的大小应用的是 disk_total_space()函数；获取磁盘分区的剩余空间应用的是 disk_free_space()函数。

disk_total_space()函数：获取一个目录的磁盘总大小，语法如下：

```
float disk_total_space ( string directory )
```

该函数根据参数 directory 提供的一个目录字符串，返回相应的文件系统或磁盘分区的所有字节数。

注意：

本函数不能作用于远程文件，其检测的文件必须通过服务器的文件系统访问。

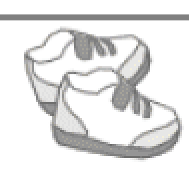
disk_free_space()函数：获取一个目录的可用空间，语法如下：

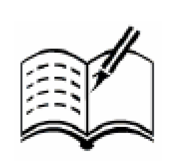
```
float disk_free_space(string directory)
```

参数 directory 用来指定文件系统或磁盘分区。

说明：

本函数返回的是该目录所在的磁盘分区的总大小，因此在给出同一个磁盘分区的不同目录作为参数所得到的结果完全相同。







应用示例

本示例应用 disk_total_space()函数获取磁盘分区大小。其具体代码如下：

（1）创建 index.php 文件。添加表单，设置文本框、提交按钮，使用 POST 方法，将数据提交到本页。

（2）通过$_POST 获取表单提交的目录路径，首先，判断获取的目录是否合理。然后，通过 iconv()函数对获取的字符串进行编码转换。接着，对获取的字符串进行截取，获取该目录所在的磁盘分区，并应用 disk_total_space()函数获取磁盘分区的大小。最后，应用 opendir()函数和 readdir()函数读取提交目录下的内容。index.php 的关键代码如下：

```
<?php
if ($_POST ['file_name'] != "" && is_file ( $_POST ['file_name'] ) == false) {
        $file_name = iconv ( "utf-8", "gb2312", $_POST ['file_name'] ); //编码格式转换
        if (file_exists ( $file_name )) {                                //判断目录是否存在
            $len = stripos ( $file_name, ":" );                          //截取字符串
            $dir = substr ( $file_name, 0, $len + 1 );                   //获取提交目录所在磁盘
            $filesize_z=disk_total_space ( $dir );                       //获取目录总大小
            $filesize_z=number_format($filesize_z/(1024*1024*1024),2,".",""); //数字的格式化
            $filesize_s=disk_free_space ( $dir );                        //获取磁盘剩余空间
            $filesize_s=number_format($filesize_s/(1024*1024*1024),2,".",""); //字节的格式化
            echo "本地磁盘（" . $dir . "）  总大小：" . $filesize_z ." GB   可用空间：
".$filesize_s." GB <br><br>";
            echo $_POST ['file_name'] . " 目录下的内容：" . "<br>";
            $i = 0;
            $list = opendir ( $file_name );                              //打开目录
            while ( $read_list = readdir ( $list ) ) {                   //读取目录
                $i ++;
                echo "$i:". iconv ("gb2312", "utf-8", $read_list ). " <br> ";  //输出目录中的内容
            }
            closedir ( $list );                                          //关闭目录
        } else {
            echo "<script>alert('目录不存在！');</script>";
        }
    } else {
        echo "<script>alert('请输入正确的目录路径！');</script>";
    }
    ?>
```

运行结果如图 7.2 所示。

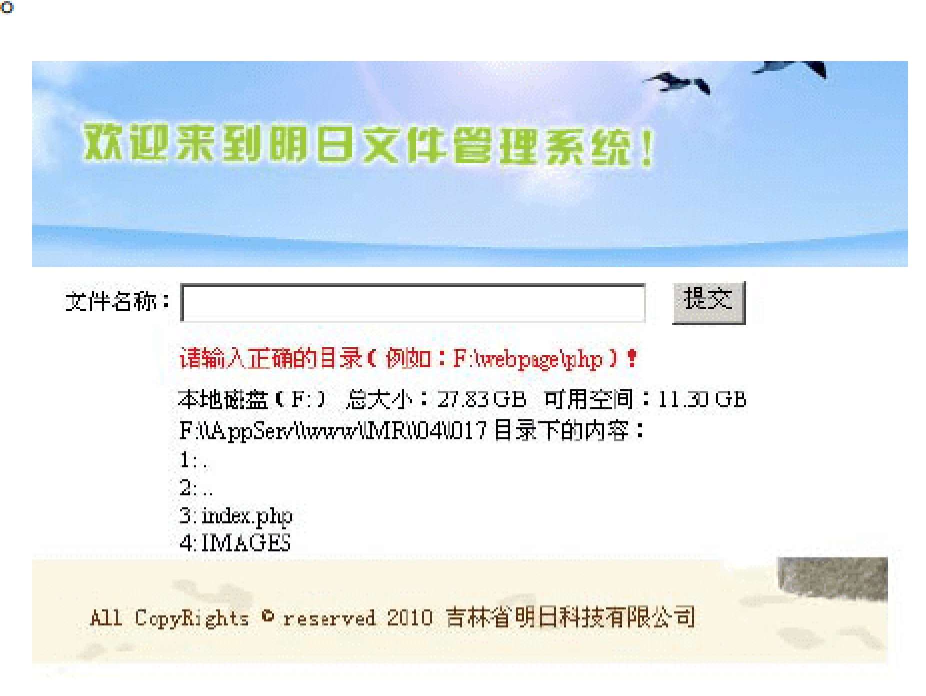
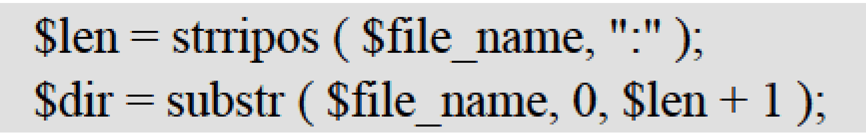


图 7.2　获取磁盘分区总大小



专家点评

strripos()函数用于判断指定字符串在另一字符串中最后出现的位置。通过本函数可以获取到指定字符串 A 在另一字符串 B 中最后出现的位置，其返回值为 int 型，根据该返回值就可以对字符串 B 以字符串 A 为分隔点进行截取，在执行截取的操作时需要应用到 substr()函数。



```
$len = strripos ( $file_name, ":" );            //截取字符串
$dir = substr ( $file_name, 0, $len + 1 );      //获取提交目录所在磁盘
```

问题 161　如何遍历指定目录下的所有文件？

问题阐述

在网站的后台管理系统中，经常需要对网站服务器中的文件进行管理和维护，有时需要添加一个文件夹、删除某个文件夹或文件，为了可以更好地查看到这些文件或文件夹，就需要对服务器指定目录下的文件进行遍历。如何实现这个指定目录下文件的遍历功能呢？

专家解答

遍历指定目录下的所有文件的关键是目录和文件处理函数的应用，包括 file_exists()、is_dir()、opendir()、readdir()和 closedir()函数。

1. 应用函数解析

（1）file_exists()：检查文件或目录是否存在。语法如下：

```
bool file_exists(string filename)
```

如果由 filename 指定的文件或目录存在，则返回 true，否则返回 false。

（2）is_dir()：判断给定的文件名是否是一个目录。

```
bool is_dir(string filename)
```

如果文件名存在并且为目录，则返回 true。如果 filename 是一个相对路径，则按照当前工作目录检查其相对路径。

说明：

本函数的结果会被缓存，且其不能作用于远程文件，被检查的文件必须通过服务器的文件系统访问。

（3）opendir()：打开目录句柄。语法如下：

```
resource opendir ( string path)
```

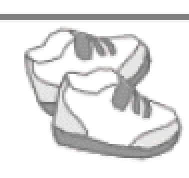
本函数返回一个目录句柄，可以在 closedir()、readdir()之前调用。如果 path 不是一个合法的目录或者因为权限限制或文件系统错误而不能打开目录，opendir()返回 false 并产生一个错误信息。

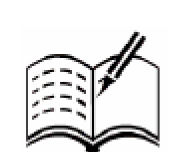
（4）readdir()：从目录句柄中读取条目。语法如下：

```
string readdir( resource dir_handle)
```

本函数返回目录中下一个文件的文件名。文件名以在文件系统中的排序返回。

Note





Note

（5）closedir()：关闭目录句柄。语法如下：

```
string closedir( resource dir_handle)
```

本函数关闭由 dir_handle 指定的目录流，目录流必须被 opendir()打开过。

综合运用上述函数，就可以实现指定目录下的文件遍历操作。

2. 应用示例

在本示例中应用上述介绍的函数，实现对指定目录下文件的遍历操作。其具体步骤如下：

（1）创建 index.php 文件。添加一个表单，定义表单元素，通过 post 方法提交目录的路径。

（2）在 index.php 中，判断提交的目录路径是否为空，如果不为空，则打开目录，读取目录中的文件，并输出读取到文件，关闭目录；否则，输出目录不存在。其关键代码如下：

```
<?php
if(!file_exists ( $look_file )){                                    //检测指定的目录是否存在
        print $look_file."目录不存在!" ;
}else{
        $i = 0;
    if( is_dir ( $look_file )){                                     //检测是否是合法目录
        if ($list = opendir ( $look_file )){                        //打开目录
            while ($read_list = readdir( $list )){                  //读取目录
                $i++ ;
            echo " $i:".iconv("gb2312","utf-8",$read_list)." <br>";  //输出目录中的内容
            }
        }
    }
    closedir ( $list ) ;                                            //关闭目录
}
?>
```

运行本示例，在文本框中输入一个指定的文件夹，单击“提交”按钮，如果该文件夹存在，就可以显示出该文件夹下包括的所有文件。运行结果如图 7.3 所示。



图 7.3　遍历指定目录下的所有文件

问题 162　如何应用 fopen()函数的选择模式？

问题阐述

如何应用 fopen()函数的选择模式？





专家解答

1. fopen 函数

在 PHP 中使用 fopen()函数打开一个文件，语法如下：

```
resource fopen (string filename, string mode [, int use_include_path [, resource zcontext]])
```

参数 filename 指定打开的文件名。

注意：

参数 filename 可以是包含文件路径的文件名（例如“C:/windows/php.ini”或“./php.ini”）。为了避免不同系统之间切换可能带来的麻烦，采用“/”作为路径分隔符。参数 filename 也可以是由某种协议给出的 URL（例如“http://mrbccd.cn”或“ftp://www.mrbccd.cn//”）。如果指定 URL 地址，则可以打开远程文件。

参数 mode 设置打开文件的方式，参数值如表 7.2 所示。

表 7.2　fopen()中参数 mode 的可选值

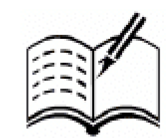
mode	模 式 名 称	说　　明
r	只读	读模式——读文件，文件指针位于文件头部
r+	只读	读写模式——读、写文件，文件指针位于文件头部。注意：如果在现有文件内容的末尾之前进行写入就会覆盖原有内容
w	只写	写模式——写入文件，文件指针指向文件头部。注意：如果文件存在，则文件内容被删除，重新写入；如果文件不存在，则函数自行创建文件
w+	只写	写模式——读、写文件，文件指针指向文件头部。注意：如果文件存在，则文件内容被删除，重新写入；如果文件不存在，则函数自行创建文件
x	谨慎写	写模式打开文件，从文件头部开始写入。注意：如果文件已经存在，函数返回FALSE，产生一个E_WARNING级别的错误信息
x+	谨慎写	读/写模式打开文件，从文件头部开始写入。注意：如果文件已经存在，函数返回FALSE，产生一个E_WARNING级别的错误信息
a	追加	追加模式打开文件，文件指针指向文件尾部。注意：如果文件已有内容，则将从文件末尾开始追加；如果文件不存在，则函数自行创建文件
a+	追加	追加模式打开文件，文件指针指向文件尾部。注意：如果文件已有内容，则从文件末尾开始追加或读取；如果文件不存在，则函数自行创建文件
b	二进制	二进制模式——用于与其他模式进行连接。注意：如果文件系统能够区分二进制文件和文本文件，可能会使用它。Windows可以区分；UNIX则不区分。推荐使用这个选项，便于获得最大程度的可移植性。它是默认模式
t	文本	用于与其他模式的结合。该模式只是Windows下的一个选项。

参数 use_include_path 为可选参数，决定是否在 include_path（php.ini 中的 include_path 选项）定义的目录中搜索 filename 文件。例如，在 php.ini 文件中设置 include_path 选项的值为“E:\AppServ\www\MR\Instance\07\”，如果希望服务器在这个路径下打开所指定的文件，则设置参数 use_include_path 的值为 1 或 TRUE。

参数 context 称为上下文，同样为可选参数，它是设置流操作的特定选项，用于控制流的







操作特性。一般情况下只需使用默认的流操作设置，不需要使用此参数。

2. 模式选择

不同的操作系统具有不同的行结束习惯。当要在文本文件中插入一个新行时，对于行结束符的选择必须适合用户的操作系统，如果使用错误的行结束符，那么当打开文件时，输出的可能是一堆乱码。所以正确地使用行结束符是非常重要的。

基于 UNIX 的系统使用\n 作为行结束符，基于 Windows 的系统使用\r\n 作为行结束符，基于 Macintosh 的系统使用\r 作为行结束符。

在 Windows 下提供一个文本转换标记（'t'）可以将\n 转换为\r\n；或者也可以使用（'b'）设置二进制模式，这样就不会对数据进行转换。所以，在 Windows 系统要使用行结束符有两个选择：一使用文本转换标记（'t'）进行转换；二使用（'b'）设置 mode 参数的模式为二进制。

默认的转换模式依赖于 SAPI 和使用的 PHP 版本。因此为了便于移植，建议在应用 fopen()函数时要指定恰当的模式。

如果想让以\n 作为行结束符的纯文本文件在其他应用程序（如 Notepad）中也有效，那么建议在在脚本中使用（'t'）模式，在其他情况下使用（'b'）。

如果在操作二进制文件时没有指定（'b'）模式，可能会输出一些奇怪的内容。因此，考虑到程序的可移植性，强烈建议在用 fopen()打开文件时总是使用（'b'）模式；同时，建议重写那些依赖于（'t'）模式的代码，使其使用正确的行结束符和正确的（'b'）模式。

问题 163　如何锁定文件？

问题阐述

在向一个文本文件写入数据时，需要先锁定文件，以防其他用户同时也修改此文件内容。如何锁定文件？

专家解答

PHP 中通过 flock()函数实现文件锁定，语法如下：

```
bool flock ( int handle, int operation)
```

参数 handle 是一个已经打开的文件指针；参数 operation 控制锁定权限，有 4 个可选值：

LOCK_SH：取得共享锁定（读取程序）。

LOCK_EX：取得独占锁定（写入程序）。

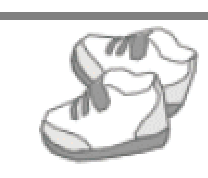
LOCK_UN：释放锁定。

LOCK_NB：防止 flock()在锁定时堵塞。

问题 164　如何访问远程文件？

问题阐述

如何访问远程文件？





专家解答

在 PHP 中访问远程文件使用 fopen()函数，前提是将配置文件 php.ini 中的选项 allow_url_fopen 设置为 ON。allow_url_fopen 参数默认是开启的，允许打开 HTTP 协议和 FTP 协议指定的远程文件，如果 allow_url_fopen 设置为 OFF，则不允许打开远程文件，fopen()函数将返回 false。

例如，首先将 php.ini 中的选项 allow_url_fopen 设置为 OFF，然后应用 fopen()函数访问一个远程文件，最后通过 readfile()函数读取文件内容。代码如下：

```
<?php
    //访问远程文件
    $filename = "http://192.168.1.249/admin/file/20080330924147598412.txt";   //要读取的文件
    $fp = fopen($filename,"rb");                                                //打开文件
?>
```

由于 php.ini 中的选项 allow_url_fopen 设置为 OFF，所以此时访问远程文件失败，将返回如图 7.4 所示的错误信息。

```
Warning: fopen(): http:// wrapper is disabled in the server
configuration by allow_url_fopen=0 in
F:\PkhPHP\www\MR\Instance\11\2.php on line 4 Warning: fopen
(http://192.168.1.249/admin/file/20080330924147598412.txt):
failed to open stream: no suitable wrapper could be found in
F:\PkhPHP\www\MR\Instance\11\2.php on line 4
```

图 7.4　访问远程文件失败

问题 165　如何读取本地文件？

问题阐述

读取本地文件的方法有哪些？

专家解答

读取文件主要是 3 种功能：读取文件到数组中、读取文件到字符串和直接将文件读取到屏幕。

1. file 函数读取文件到数组中

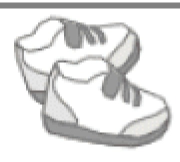
file()函数将整个文件的内容读入到一个数组中。成功返回数组，数组中的每个元素都是文件中对应的一行，包括换行符在内；失败返回 FALSE。语法如下：

```
array file ( string filename [, int use_include_path [, resource context]] )
```

参数 filename 指定读取的文件名称；参数 use_include_path 控制是否支持在 include_path 中搜索文件，如果支持，则将该值设置为 TRUE；参数 context 是 PHP5.0 新增内容。

PHP 的 file()函数将文件读取到数组中，在读取过程中使用新行符指示新数组元素从哪里开始。其具体应用如下：

```
<?php
$file=file('zhuce.php');
$lines=count($file);
$alt='';
for($i=0; $i<$lines;$i++){
```




```
        $alt=($alt=='even')?'odd':'even';
        echo'<div class="'.$alt.'">';
        echo $i.':'.htmlspecialchars($file[$i]);
        echo"</div>\n";
    }
?>
```

以上程序将文件输出到一个格式良好的页面中，因此可以逐行地进行检查。使用 for 循环读取$file 数组变量，并按照希望的格式显示。在 for 循环语句中应用三元运算符设置行的颜色。

2. file_get_contents 函数读取文件到字符串中

file_get_contents()函数将文件内容读入一个字符串。如果有 offset 和 maxlen 参数，将在参数 offset 所指定的位置开始读取长度为 maxlen 的内容。如果失败，则返回 FALSE。语法如下：

```
string file_get_contents ( string filename [, bool use_include_path [, resource context [, int offset [, int maxlen]]]] )
```

参数 filename 指定读取的文件名称；参数 use_include_path 控制是否支持在 include_path 中搜索文件，如果支持，则将该值设置为 TRUE。

PHP 从 4.3 版本开始包含 file_get_contents 的函数，该函数能够在不分解文件的情况下直接将文件读取到字符串中。其具体应用如下：

```
<?php
$context = array(
    'http' => array(
        'proxy' => 'tcp://127.0.0.1',
        'request_fulluri' => true,
        ),
    );
$contexts = stream_context_create($context);
$files = file_get_contents("http://www.mrbccd.com", false, $contexts);
echo $files;
?>
```

3. readfile 函数直接读取文件

readfile()函数读取一个文件并写入输出缓冲，如果成功返回则读取字节数，如果失败则返回 FALSE。语法如下：

```
int readfile ( string filename [, bool use_include_path [, resource context]] )
```

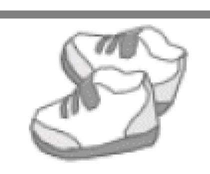
参数 filename 指定读取的文件名称；参数 use_include_path 控制是否支持在 include_path 中搜索文件，如果支持，则将该值设置为 TRUE；参数 context 是 PHP 5.0 的新增内容。

注意：

应用 readfile()函数，不需要打开/关闭文件，不需要输出语句，直接应用函数即可。

专家点评

Readfile 是一种保护文件和带宽的便捷方法。通过一个使用 readfile 函数的脚本将用户 Web 站点中的所有文件链接起来，能够防止其他站点直接链接到它们，从而阻止其对 Web 站点带宽的可能占用。这种方法使用的技术通常被称为“反盗链”。如果引入了验证系统和/或 HTTP 引





用检查，就拥有了一个确保只有合法参观者能够访问您文件的安全系统。

问题 166　如何使用文件句柄？

问题阐述

为了使用读取文件函数，需要使用相对于 PHP 脚本的路径为它们指出要读取的文件。然而，大多数 PHP 的文件函数使用一种稍有区别的机制访问文件，这种机制与连接数据库的机制非常相似。该过程使用 fopen 函数进行“连接”，使用 fclose 函数“断开连接”。fopen 函数的返回值是一个 PHP 文件指针，也被称为文件句柄。一旦拥有了文件句柄，就可以使用它对文件执行多种操作，包括读取、附加操作等。那么这个句柄在使用过程中都需要注意哪些问题呢？

专家解答

首先来看一下文件是如何打开和关闭的。其代码如下：

```
<?php
    $f_open =fopen("../file.txt.","rb");          //打开文件
    …                                             //对文件进行操作
    fclose($f_open)                               //操作完成后关闭文件
?>
```

使用 fopen 连接文件时，必须指定文件路径和文件访问模式（例如，r 表示只读模式）。b 模式表示文件将以二进制模式打开，二进制模式可以用来保证代码在不同操作系统之间具体可移植性。

1. 通过句柄定位文件读取的起始位置

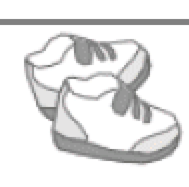
在通过 fopen()函数打开一个文件值时就已经拥有了一个文件句柄，下面就应用该句柄读取文件。其代码如下：

```
<?php
    $filename = "04.txt";                         //要读取的文件
    $fp = fopen($filename,"rb");                  //打开文件
    echo fread($fp,32);                           //使用 fread()函数读取文件内容的前 32 个字节
    echo "<p>";
    echo fread($fp,filesize($filename));          //输出其余的文件内容
?>
```

这个例子演示了如何使用文件句柄。注意，当使用 fread 时，第二个参数以字节为单位，表示从文件开始读取的数据量。文件内容被读取到设置的数据量时，句柄将定位到这个位置，如果想要继续读取这个文件中的其他内容，则可以通过 filesize 函数获取文件的总大小，并且将这个值作为 fread 的第二个参数，此时输出的将是文件剩余部分的内容。这就是句柄的作用。

2. 大文件的分块读取

前面的方案对于小文件读取非常有效，但是在读取较大文件的内容时，服务器的内存将被占用，由此可能会造成性能问题。为了减少发生该问题的可能，可以采用以块读取文件的方法，并以块为对象进行操作，这样就避免了对内存的占用。其原理是，通过 fopen 函数打开文件，





应用 while 语句循环读取文件的内容，当文件指针到达文件结束位置时停止循环；否则应用 fgets 函数读取文件中数据，并且向前移动文件句柄。以块读取文件的具体代码如下：

```
<?php
        $fopen = fopen('fun.php','rb');
        while(!feof($fopen)){                    //使用 feof()函数测试指针是否到了文件结束的位置
            echo fgets($fopen);                  //输出当前行
        }
    fclose($fopen);
?>
```

专家点评

还有很多函数能够使用文件句柄读取文件。其中一个是 fgetss 函数（注意是两个 s），它除了能够像 strip_tags 函数那样剥离其发现的所有 HTML 标记之外，其他方面几乎与 fgets 相同。另一个是 fscanf 函数，它像 printf 一样格式化来自文件的输出。另外还有 fgetcsv，它使得处理 csv（comma separated values，逗号分割值）文件变得十分容易。

但是如果希望将文件的全部内容都读取到一个变量中，那么 file 和 file_get_content 函数更易于使用，它们还潜在地提供了更好的性能。

问题 167　如何在文件中存储配置信息?

问题阐述

某些在站点中重复使用的信息（如密码、路径和变量）最好存储在一个文件中。这样，如果需要把代码转移到另一个站点，就只需要修改一次设置，而不用成百上千次地修改代码。那么如何将这些配置信息存储到指定的文件中的呢？又如何读取这些配置信息呢？

专家解答

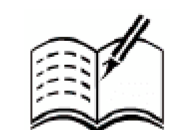
最简单的存储配置信息的方法是在一个.ini 文件中建立变量，然后使用 parse_ini_file 函数在代码中包含这个文件，该函数能够解析与 php.ini 相同格式的文件。例如，下面建立一个 config.ini 文件。其内容如下：

```
; This is a mysql configuration file
; Comments start with ';', as in php.ini
host = localhost
dbname = db_database
username = root
password = 111
charset = utf8
```

应用 parse_ini_file 函数读取配置文件 config.ini 中的数据，其代码如下：

```
<?php
$ini_array = parse_ini_file("config.ini");
print_r($ini_array);
$ini_array = parse_ini_file("config.ini", true);
print_r($ini_array);
?>
```





输出结果如下：

```
Array
(
    [host] => localhost
    [dbname] => db_database
    [username] => root
    [password] => 111
    [charset] => utf8
)
Array
(
    [host] => localhost
    [dbname] => db_database
    [username] => root
    [password] => 111
    [charset] => utf8
)
```

专家点评

在存储配置文件时，最好不要将配置文件存储于 Web 根目录下，因为配置文件通常会包含用户名和密码信息。如果将配置文件存储于 Web 根目录下，则很容易被“不速之客”访问。所以，还是应将其存储于服务器之外，这样“不速之客”就很难访问这个文件了。

如果必须将配置文件存储到 Web 根目录下，一定要在.htaccess 文件中包含一条文件指令，用于限制能够访问该类文件的用户。

问题 168　如何在 PHP 中创建压缩的 RAR 文件？

问题阐述

在网站中，上传的文件大多数都是以压缩包的形式存在的，所以先将要上传的文件压缩后才能够上传，那么如何实现网站中上传文件的自动压缩功能呢？

专家解答

文件压缩主要应用 PHP 的预定义类 com，将 wscript.shell 对象作为 com 类的参数，最终通过调用 com 类的 run()方法实现文件的压缩。

com 类提供一个将 COM 组件整合到 PHP 脚本中的框架。实例化方法如下：

```
$obj = new COM("server.object")
```

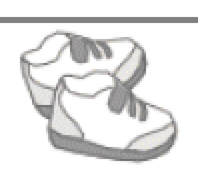
com 类构造函数的语法如下：

```
string COM::COM ( string module_name [, string server_name [, int codepage]] )
```

参数说明：

❶ module_name：被请求组件的名字或 class-id。

❷ server_name：DCOM 服务器的名字，组件在此服务器上被取用。如果是 NULL，则假定是 localhost。如果要支持允许 DCOM，则必须将 php.ini 中的 com.allow_dcom 设置为 TRUE。





Note

❸ codepage：指定用于将 PHP 字符串（php-strings）转换成 UNICODE 字符串（unicode-strings）的代码页，反之亦然。可用的值有 CP_ACP、CP_MACCP、CP_OEMCP、CP_SYMBOL、CP_THREAD_ACP, CP_UTF7 和 CP_UTF8。

com 类中 run()方法的语法如下：

```
run("winrar a  生成的 rar 文件的路径及名称  要压缩的文件的路径及名称",1,false);
```

应用示例

本示例中讲解在 PHP 中如何创建 RAR 文件。其具体步骤如下：

（1）创建 index.php 文件。首先，创建一个表单，设置文件域，提交要压缩的文件；设置文本框，提交压缩文件的存储位置；设置按钮，完成表单元素的提交操作。

（2）在 index.php 文件中编写 PHP 脚本，通过$_POST[]方法获取表单提交的数据，通过 iconv()函数完成数据编码格式的转换，实例化 com 类，调用 run()方法完成文件的压缩操作。其关键代码如下：

```
<?php
    if ($_POST ['Submit'] != "") {
        $sourcefile = iconv ( "utf-8", "gb2312", $_POST ['sourcefile'] );    //获取指定文件的路径
        $objfile = $_POST ['objfile'] . "\\" . time ();              //定义压缩文件的存储位置和名称
        $objfile = iconv ( "utf-8", "gb2312", $objfile );            //实现编码转换
        $obj = new com ( "wscript.shell" );                          //实例化 com 类
        $obj->run ( "winrar a " . $objfile . " " . $sourcefile . "", 1, false );   //调用类中的方法
        echo "<script>alert('文件压缩完成!');</script>";
    }
    ?>
```

运行本示例，首先选择要压缩的文件，然后设置压缩文件的存储位置，单击“提交”按钮后。其运行结果如图 7.5 所示。

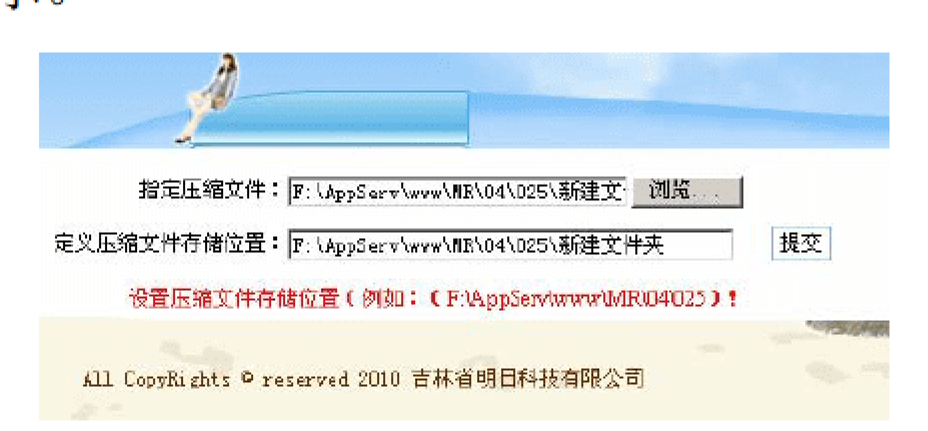


图 7.5　RAR 压缩文件

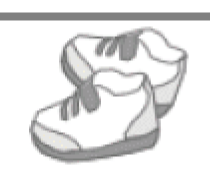
专家点评

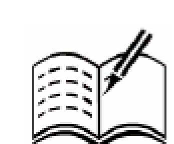
在执行文件压缩操作时，切记在文件的存储路径中不要出现空格（包括被压缩文件的路径、名称以及压缩文件的存储路径），否则压缩操作将不会正确执行。

问题 169　如何在 PHP 中对 RAR 文件进行解压?

问题阐述

在 PHP 中既然可以对文件进行压缩，那么就一定可以对这个压缩的文件进行解压，这个解压的功能又是如何实现的呢？





专家解答

文件解压同样应用 PHP 的预定义类 com，将 wscript.shell 对象作为 com 类的参数，通过调用 com 类的 run()方法执行文件解压的命令。文件解压的命令如下：

```
$obj->run ( "winrar x -av -c ".$sourcefile." ".$objfile. "", 1, true ); //调用类中的方法，执行文件解压的命令
```

通过此命令将指定的压缩文件解压到指定的绝对路径下。

应用示例

本示例对压缩的 RAR 文件进行解压。其具体步骤如下：

（1）创建 index.php 文件。首先，创建一个表单，设置文件域，提交要解压的文件；设置文本框，提交解压后文件的存储位置；设置按钮，完成表单元素的提交操作。

（2）在 index.php 文件中编写 PHP 脚本，通过$_POST[]方法获取表单提交的数据，通过 iconv()函数完成数据编码格式的转换，实例化 com 类，调用 run()方法执行文件的解压命令。其关键代码如下：

```
<?php
if ($_POST ['Submit'] != "") {
        include("function.php");
        $sourcefile=iconv("utf-8", "gb2312", unhtml($_POST['sourcefile'])); //获取指定文件的路径
        $objfile=iconv( "utf-8", "gb2312", $_POST ['objfile'] );//定义解压文件的存储位置和名称
        $obj = new com( "wscript.shell" );                    //实例化 com 类
        $obj->run ( "winrar x -av -c ".$sourcefile." ".$objfile. "", 1, true );  //调用类中的方法，执行
                                                                          解压命令
        echo "<script>alert('文件解压完成!');</script>";
    }
?>
```

运行本示例，首先选择要解压的 RAR 文件，并在文本框中输入要解压文件的存储目录，然后单击“解压”按钮即可。其运行结果如图 7.6 所示。



图 7.6　解压 RAR 文件

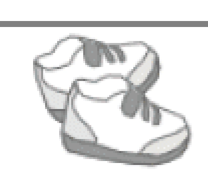
专家点评

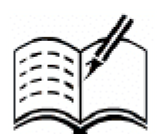
在执行文件解压操作时，切记在文件的存储路径中不要出现空格（包括被解压文件的路径、名称以及解压后文件的存储路径），否则解压操作将不会正确执行。

问题 170　如何将文本文件中的数据存储到数据库中？

问题阐述

如何将文本文件中的数据存储到数据库中？





专家解答

将文本文件中的数据存储到数据库中主要运用到 3 方面的技术：第一是将文本文件上传到服务器；第二是读取服务器指定文件夹下的文本文件中的数据；第三是连接数据库，将从文本文件中读取的数据存储到数据表中。

其中，最为关键的是将文本文件中的数据存储到数据库中，在执行数据的读取和存储时，有一个细节必须把握。即应用 addslashes()函数，对从文本文件中读取的数据进行转义，因为在读取的数据中可能包含单引号（'）、双引号（"）、反斜线（\）或 NUL（NULL 字符），如果不对其进行转换，则在执行 SQL 语句时可能会出现错误。

addslashes()函数通过反斜线来引用字符串。语法如下：

```
string addslashes ( string str)
```

在其返回的字符串中，为了数据库查询语句等的需要在某些特定字符前加上了反斜线。这些特定字符包括单引号（'）、双引号（"）、反斜线（\）和 NUL（NULL 字符）。

说明：

默认情况下，PHP 指令会自动对所有的 GET、POST 和 COOKIE 数据运行转义（addslashes()函数）。所以，如果是通过 GET 或 POST 方法提交的数据是不需要进行转义的，可以直接添加到数据库中；如果提交的数据是从文本文件中读取或者自定义的内容，那么就有必要对其进行转义。

应用示例

本示例将文本文件中的数据存储到数据库中，其具体实现步骤如下：

（1）创建 index.php 文件。首先，设计网页页面。然后，创建一个表单，通过文件域提交要上传的 ini 文件。接着，获取表单中提交的文件，将文件存储到服务器指定的文件夹下，通过 file_get_contents()函数读取上传文件的内容，并且将读取的结果存储到指定的数据表中。最后，执行查询语句，输出数据库中存储的 ini 文件的内容。其关键代码如下：

```
<?php
if (! empty ( $_FILES [up_picture] [name] )) {                    //判断上传内容是否为空
    $type = $_FILES ['up_picture'] ['name'];
    $types = strstr ( $type, '.' );                               //获取文件后缀
    if ($types == ".ini") {
        if ($_FILES ['up_picture'] ['error'] > 0) {               //判断文件是否可以上传到服务器
            echo "上传错误:";
            switch ($_FILES ['up_picture'] ['error']) {
                case 1 :
                    echo "上传文件大小超出配置文件规定值";
                    break;
                case 2 :
                    echo "上传文件大小超出表单中约定值";
                    break;
                case 3 :
                    echo "上传文件不全";
                    break;
                case 4 :
```





```
                        echo "没有上传文件";
                        break;
                }
            } else {
                if (! is_dir ( "./upfile/" )) {                  //判断指定目录是否存在
                    mkdir ( "./upfile/" );                        //创建目录
                }
                $file_name = time () . strstr ($_FILES ['up_picture'] ['name'], '.'); //定义上传文件名称
                $path = './upfile/' . $file_name;                //定义上传文件名称和存储位置
                if (is_uploaded_file ( $_FILES ['up_picture'] ['tmp_name'] )) {
                                                                  //判断文件是否是 HTTP POST 上传
                    if (! move_uploaded_file ( $_FILES ['up_picture'] ['tmp_name'], $path )) {
                                                                  //执行上传操作
                        echo "上传失败";
                    } else {
                        //读取文本文件中的数据，并且实现转义和编码的转换
                        $arr = iconv ( "gb2312", "utf-8", addslashes ( file_get_contents ( $path ) ) );
                        include_once ("conn/conn.php");                  //连接数据库
                        $sql = "insert into tb_files(file_name,file_content,file_date)values('$file_name',
'$arr','" . date ( "Y-m-d H:i:s" ) . "')";                              //定义 SQL 添加语句
                        $result = mysql_query ( $sql, $conn );          //执行添加操作
                        echo "文件：  " . $_FILES ['up_picture'] ['name'] . "  上传成功!" . "<br>";
                        echo "大小：  " . $_FILES ['up_picture'] ['size'] . "  字节";
                    }
                } else {
                    echo "上传文件" . $_FILES ['up_pictute'] ['name'] . "不合法！ ";
                }
            }
        } else {
            echo "上传文件" . $_FILES ['up_pictute'] ['name'] . "类型不正确！ ";
        }
    }
    ?>
```

（2）创建 conn 文件夹，编写 conn.php 文件，连接数据库服务器，连接 db_database 数据库。其代码如下：

```
<?php
$conn=mysql_connect("localhost", "root", "111")or die("连接数据库服务器失败！ ". mysql_error () );
mysql_select_db ( "db_database", $conn );                    //选择数据库 db_database
mysql_query ( "set names utf8" );                            //设置数据库编码格式 utf8
?>
```

运行结果如图 7.7 所示。

专家点评

将从文本文件中读取的数据存储到数据库时，必须应用 addslashes()函数对文本文件返回的字符串进行转义，否则有可能出现如图 7.8 所示的错误。

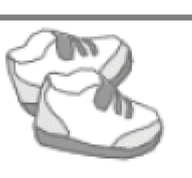




图 7.7　将文本文件中的数据存储到数据库中

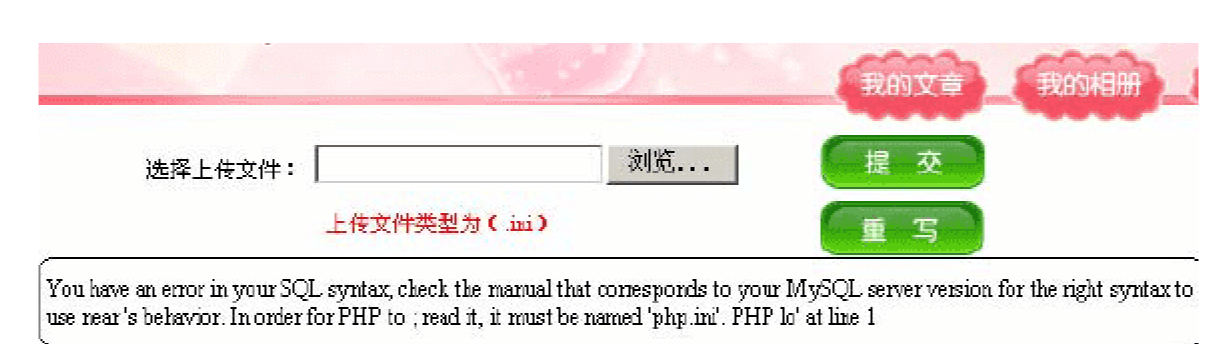


图 7.8　数据添加失败

问题 171　如何判断一个远程的文件是否存在？

问题阐述

在 PHP 中，如何判断一个远程的文件是否存在呢？

专家解答

判断一个远程的文件是否存在与访问一个远程文件应用的都是 fopen 函数。如果通过 fopen 函数可以访问指定的远程文件，并且返回标识指针，则说明该远程文件存在，否则说明该远程文件不存在。

应用 fopen()函数判断远程文件是否存在，前提是将配置文件 php.ini 中的选项 allow_url_fopen 设置为 ON。allow_url_fopen 参数默认是开启的，允许打开 HTTP 协议和 FTP 协议指定的远程文件，如果 allow_url_fopen 设置为 OFF，则不允许打开远程文件，fopen()函数将返回 false。

注意：

在实际的生产环境中为了保证服务器的安全，php.ini 中选项 allow_url_fopen 的值一般情况下都设置为 OFF，所以不支持 fopen()函数访问远程文件，这一点必须明确。

问题 172　如何将上传文件转换成 RAR 文件？

问题阐述

在 PHP 中，如何实现将上传文件转换成 RAR 文件？

专家解答

将上传文件转换成 RAR 文件，可以理解为两种技术的结合应用，首先是文件上传技术，然后是文件压缩技术。

文件上传技术自不必说，文件压缩技术是调用 com 类中的 run()方法，在 winrar 的命令中增加了两个属性："-ep"将压缩文件以文件名的格式进行存储，不包括文件的路径；"-df"文件压缩成功后删除原始文件。其完整的命令语句如下：

```
$obj->run ( "winrar a -ep -df " . $file_path.".rar" . " " . $path . "", 1, false ); //调用类中的方法，文件压缩
```



应用示例

在本示例中将上传的文件直接以压缩文件的格式进行存储。其具体步骤如下：

（1）创建 index.php 文件。首先，创建一个表单，设置文件域，提交上传文件。然后设置按钮，完成表单元素的提交操作。

（2）在 index.php 文件中编写 PHP 脚本，首先，对上传文件进行判断，如果符合要求，将上传文件存储到服务器指定文件夹下。然后，实例化 PHP 中 com 类，通过 run()方法执行文件的压缩命令，完成上传文件的压缩操作，并将原始文件删除。其关键代码如下：

```
//定义上传文件名称和存储位置
$path = './upfile/' . time () . strstr ( $_FILES ['sourcefile'] ['name'], '.' );
    if (is_uploaded_file ( $_FILES ['sourcefile'] ['tmp_name'] )) {
    if (! move_uploaded_file ( $_FILES ['sourcefile'] ['tmp_name'], $path )) { echo "上传失败";
        } else {
            $path = realpath ( iconv ( "utf-8", "gb2312", $path ) );    //获取文件在服务器中的路径
            $len = strripos ( $path, "." );                             //计算路径的长度
            $file_path = substr ( $path, 0, $len );                     //获取上传文件的路径名称
            $obj = new com ( "wscript.shell" );                         //实例化 com 类
//调用类中的方法，执行文件压缩和原始文件删除操作
            $obj->run ( "winrar a -ep -df " . $file_path . ".rar" . " " . $path . "", 1, false );
            echo "文件：" . $_FILES ['sourcefile'] ['name'] . "上传成功！" . "<br>";
            echo "大小：" . $_FILES ['sourcefile'] ['size'] . " 字节";
        }
    } else {
        echo "上传文件" . $_FILES ['sourcefile'] ['name'] . "不合法！";
    }
```

运行本示例，选择要上传的文件，单击“提交”按钮，其运行结果如图 7.9 所示。上传成功后，可以查看本实例根目录下的 upfile 文件夹，以及上传文件是否以压缩文件的格式进行存储，如图 7.10 所示。



图 7.9　将上传文件转换成压缩文件

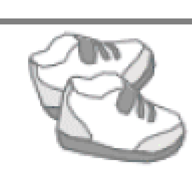


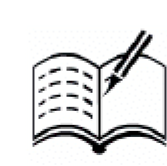
图 7.10　上传后的文件

问题 173　如何遍历、删除指定目录下的所有文件？

问题阐述

在 PHP 中，如何实现遍历及删除某一目录下的所有文件呢？





专家解答

1. 解决思路

遍历、删除指定目录下的所有文件关键是目录、文件处理函数和字符串处理函数的综合运用。

☑ 目录处理函数

（1）chdir()函数，改变当前目录。语法如下：

```
bool chdir(string directory)
```

参数 directory 用于指定更改后目录的位置。操作成功返回 True，否则返回 False。

（2）getcwd()函数，获取当前工作的目录。语法如下：

```
string getcwd(void);
```

（3）rmdir()函数，删除目录。语法如下：

```
bool rmdir (string dirname)
```

尝试删除参数 dirname 所指定的目录。该目录必须是空的，而且要有相应的权限。如果成功则返回 True，失败则返回 False。

☑ 文件操作函数

（1）fopen()函数、fgets()函数、readfile()函数和 fclose()函数。

（2）filectime()函数，返回指定文件的索引节点修改时间。语法如下：

```
int filectime(string filename);
```

返回文件上次索引节点被修改的时间，如果出错则返回 False。时间以 UNIX 时间戳的方式返回。

（3）filemtime()函数，返回指定文件 filename 的最后修改时间。语法如下：

```
int filemtime(string filename);
```

返回文件上次被修改的时间，如果失败则返回 FALSE。时间以 UNIX 时间戳的方式返回。

（4）unlink()函数，删除文件。语法如下：

```
bool unlink (string filename)
```

删除由字符串 filename 指定的文件。如果成功则返回 True，失败则返回 False。

☑ 字符串操作函数

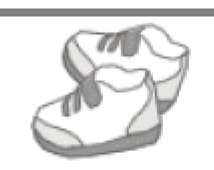
（1）substr()函数，从指定的字符串中按照指定的位置截取一定长度的字符。语法如下：

```
string substr(string str,int start,int length)
```

substr()函数的参数说明如表 7.3 所示。

表 7.3 substr()函数的参数说明

参　数	说　明
str	指定字符串对象
start	指定开始截取字符串的位置。如果参数start为负数，则从字符串的末尾开始截取
length	可选参数，指定截取字符的个数。如果length为负数，则表示取到倒数第length个字符





注意：

substr()函数中参数 start 的指定位置是从 0 开始计算的，即字符串中的第一个字符表示为 0。

（2）strrpos()函数，寻找字符串 haystack 中的字符 needle 最后出现的位置。语法如下：

```
int strrpos(string haystack, char needle)
```

参数 haystack 为寻找的范围；参数 needle 为寻找的字符，其中 needle 只能是一个字符，汉字则不可以。如果找不到指定的字符，则返回 False。

（3）strtoupper()函数，将字符串 str 全部转换为大写字符串。语法如下：

```
string strtoupper(string str);
```

（4）htmlentities()函数，将所有 string 的字符都转换成 HTML 的特殊字集字符串。语法如下：

```
string htmlentities(string string);
```

2. 应用示例

本示例综合运用上述介绍的函数，完成对网站目录的遍历以及目录和文件的删除操作。其具体步骤如下：

（1）创建 index.php 文件。应用目录、文件处理函数和字符串处理函数完成对指定目录下文件的读取操作，并设置查看文件内容的超链接和删除指定文件的超链接。在设置超链接的参数时，应用 urlencode()函数对字符串进行编码。

（2）创建 look_file.php 文件，完成对指定文件内容的读取操作。其关键代码如下：

```
<?php
$catalog = iconv ( "utf-8", "gb2312", urldecode ( $_GET [catalog] ) );  //获取文件的信息，设置编码
$filename = iconv ( "utf-8", "gb2312", urldecode ($_GET [filename])); //获取文件的信息，设置编码
$type = iconv ( "utf-8", "gb2312", urldecode ( $_GET [type] ) );        //获取文件的信息，设置编码
$types = strtoupper ( $type );                                           //完成字符串的大小写转换
if ($types == ".PHP") {                                                  //显示 PHP 文件的内容
    $arr = file ( $catalog . "\\" . $filename );                         //读取文件
    foreach ( $arr as $value ) {                                         //循环输出文件内容
        $value = htmlentities ( $value, ENT_COMPAT, "UTF-8" );           //特殊字符的转换
        echo $value . "<br>";                                            //输出内容
    }
} else {
    $fp = fopen ( $catalog . "\\" . $filename, "r" );                    //显示 txt 文件的内容
    while ( $line = fgets ( $fp ) ) {
        $line = htmlentities ( $line, ENT_COMPAT, "GB2312" );            //特殊字符的转换
        echo iconv ( "gb2312", "utf-8", $line );                         //输出文件内容
    }
    fclose ( $fp );
}
?>
```

（3）创建 delete.php 文件，完成对指定文件的删除操作。

运行本示例，实现目录和文件的遍历和删除，运行结果如图 7.11 所示。






图 7.11　遍历、删除指定目录下的所有文件

专家点评

在通过 URL 传递字符串参数时，应尽量应用 urlencode()函数对传递的参数值进行编码，这样不但可以保证传递参数值的安全，而且可以防止传递的参数值出现乱码的问题。例如，本实例中在执行指定目录、文件的删除操作过程，在执行成功跳转到上级页面时，如果不对 URL 传递的字符串参数值进行编码，那么就无法完成跳转的操作，同时会弹出一个错误提示。

问题 174　如何使用文本文件统计网站访问量？

问题阐述

网站的计数器对于网站管理者来说是一个非常值得关注的部分，它记录了网站被访问的次数，客观地反映了网站受欢迎的程度。而文本计数器是最简单的一种，它将数据存储于文本文件中。那么如何使用文本文件统计网站访问量，并将其显示到页面中呢？

专家解答

1. 解决思路

文本计数器的设计思路如下：

应用文件系统函数和 GD2 函数，完成网站访问量的存储、读取和输出。其实现的原理如图 7.12 所示。

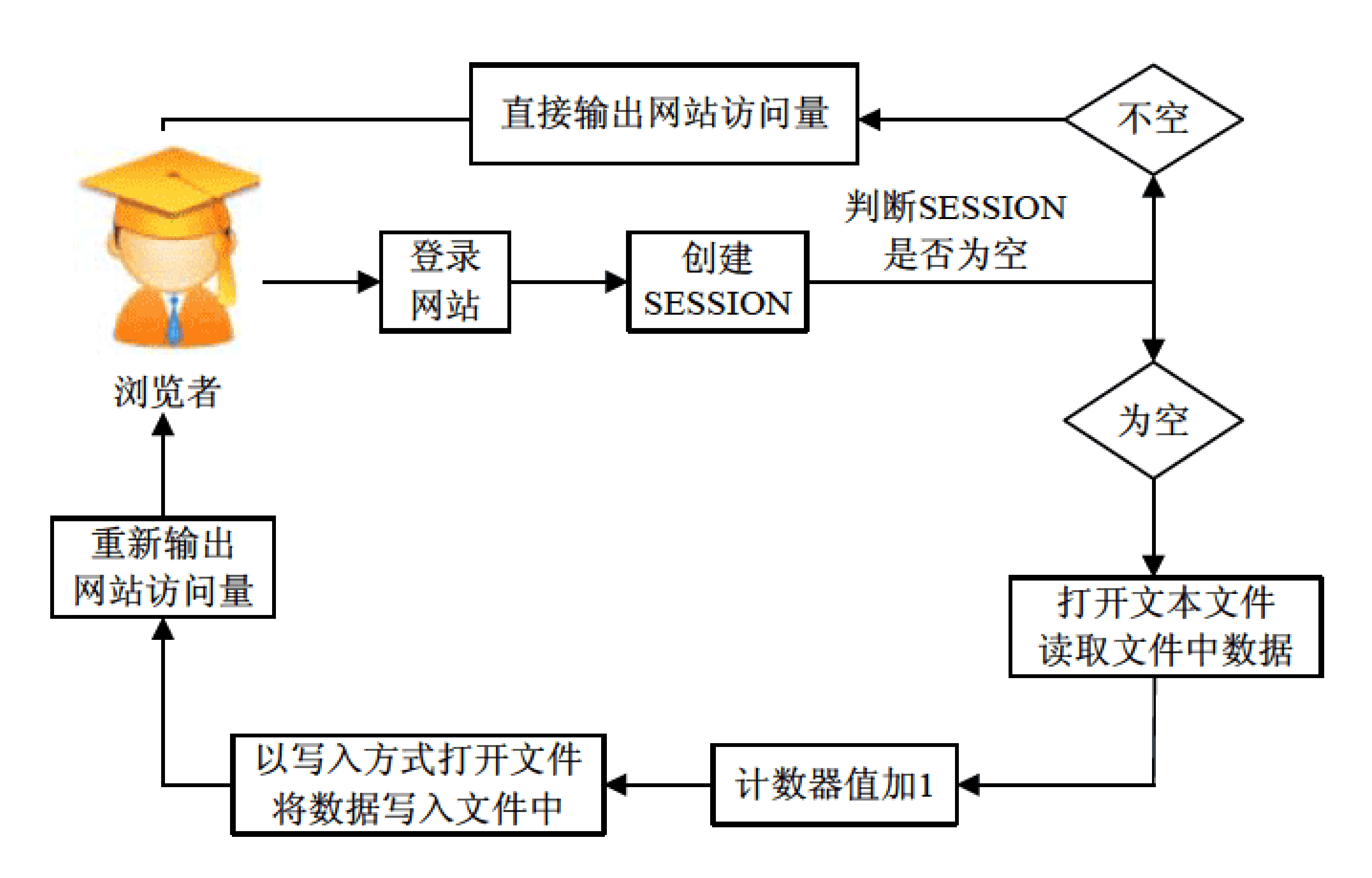
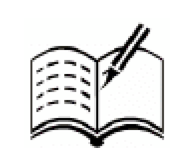


图 7.12　通过文本文件统计网站访问量设计原理





在这个通过文本文件统计网站访问量的程序中，概括起来应用了 3 个方面的技术：

第一方面，文件系统操作函数，实现数据的写入、读取功能。

第二方面，SESSION，通过 SESSION 变量防止重复计数。

第三方面，GD2 函数，将访问量数据生成图像，改变数据的输出方式。

2. 应用示例

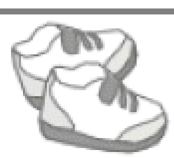
本示例应用上述介绍的技术，通过文本文件统计网站访问量，通过 SESSION 变量屏蔽页面刷新对计数器的影响，通过 GD2 函数库中的函数输出网站访问量。其具体步骤如下：

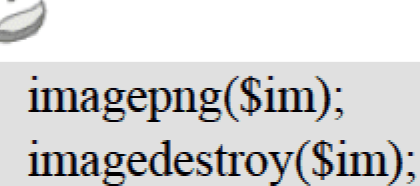
（1）创建 index.php 文件。通过文件操作函数和 SESSION 全局变量完成网站访问量的统计和防止重复计数，统计数据存储在实例根目录下的 counter.txt 中。通过 img 图像标签的 src 属性指定 gd1.php 文件，以图像的形式输出网站访问量的数据。其关键代码如下：

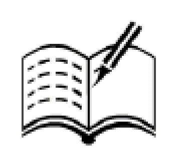
```
<?php session_start();
if($_SESSION[temp]==""){                //判断$_SESSION[temp]==""的值是否为空，其中的 temp
                                         为自定义的变量
    if(($fp=fopen("counter.txt","r"))==false){
        echo "打开文件失败!";
    }else{
        $counter=fgets($fp,1024);       //读取文件中数据
        fclose($fp);                    //关闭文本文件
        $counter++;                     //计数器增加 1
        $fp=fopen("counter.txt","w");   //以写的方式打开文本文件<!---->
        fputs($fp,$counter);            //将新的统计数据增加 1
        fclose($fp);                    //关闭文件
    }
     $_SESSION[temp]=1;                      //登录以后，$_SESSION[temp]的值不为空，给
                                              $_SESSION[temp]赋一个值 1
}
?>
<tr>
    <td height="40" align="center"><img src="gd1.php" /></td>
</tr>
```

（2）创建 gd1.php 文件，根据从文本文件 counter.txt 中读取到的数据，通过 GD2 函数生成数据图像。其代码如下：

```
<?php
if(($fp=fopen("counter.txt","r"))==false){                  //打开文件
    echo "打开文件失败!";
}else{
    $counter=fgets($fp,1024);                               //读取文件中数据
    fclose($fp);                                            //关闭文件
    $im=imagecreate(240,24);                                //通过 GD2 函数创建画布
    $gray=imagecolorallocate($im,255,255,255);              //定义背景颜色
    $color =imagecolorallocate($im,rand(0,255),rand(0,255),rand(0,255));      //定义字体颜色
    //输出中文字符
    $text=iconv("gb2312","utf-8","网站的访问量:");           //对指定的中文字符串进行转换
    $font = "Fonts/FZHCJW.TTF";                             //定义使用字体的存储位置
    imagettftext($im,14,0,20,18,$color,$font,$text);        //输出中文
    imagestring($im,5,160,5,$counter,$color);               //输出图像，输出网站的访问次数
```







Note

```
        imagepng($im);                                          //生成 png 图像
        imagedestroy($im);                                      //销毁图像
    }
    ?>
```

（3）在根目录下创建 counter.txt 文本文件，设置默认值为 0；创建 fonts 文件夹，将使用的字体 FZHCJW.TTF 存储在该文件夹下。

本示例的运行结果如图 7.13 所示。



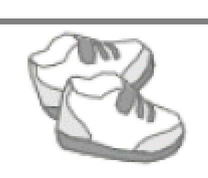
图 7.13　通过文本文件统计网站访问量

专家点评

SESSION 防止重复计数的原理是：首先，在网页被访问时，初始化一个 SESSION 变量，并赋给其一个空值。然后，判断 SESSION 变量的值是否为空，如果为空，则将计数器的值增加 1，并且为 SESSION 变量赋值为 1。此时，在当前页中，SESSION 变量的值已经不为空，无论如何刷新，SESSION 变量的值都不会改变，所以计数器的值也不会增加。



面向对象的程序设计



问题 175　为什么要面向对象？



Note

问题阐述

面向对象（OOP）的编程方式是 PHP 的突出特点之一，那么为什么要使用面向对象呢？

专家解答

要解释为什么使用面向对象，首先应知道面向对象思想的来源和它能解决的问题，下面分别进行介绍。

1. 面向对象思想的来源

面向对象思想来源于对现实世界的认知。现实世界缤纷复杂、种类繁多，难于认识和理解。但是聪明的人们学会了把这些错综复杂的事物进行分类，从而使世界变得井井有条。例如，现实生活中，由各式各样的汽车抽象出汽车的概念，由形形色色的鸟抽象出鸟的概念，由五彩斑斓的鲜花抽象出花的概念等，如图 8.1 所示。

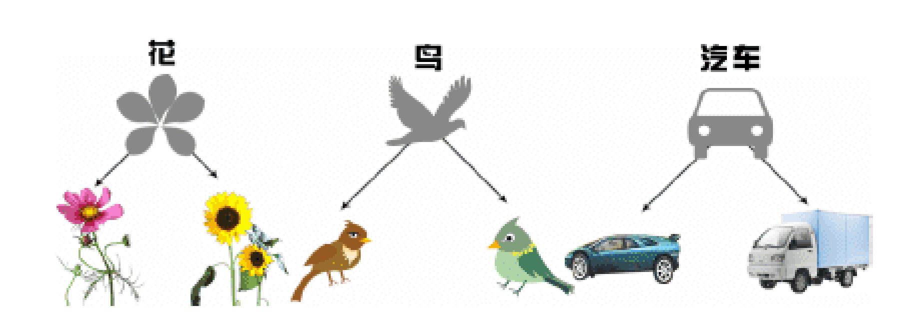


图 8.1　面向对象思想来源于对现实世界的认知

2. 面向对象能解决的问题

通过面向过程进行 Web 项目开发，一般都会存在以下问题：

☑　代码重用性差

程序开发过程中，有时会遇到实现相同功能的代码到处进行复制、粘贴的情况，存在代码重用性差的问题。

☑　项目可维护性差

项目开发过程中，项目的可读性、可修改性和可测试性是项目的重要质量指标，用传统方法开发出来的项目，维护费用和成本比较高。

☑　开发出来的项目不能满足用户的需求变化

用传统的面向过程方法开发出的项目涉及各种不同领域的知识，在开发需求模糊或需求动态变化的系统时，所开发项目往往不能满足用户需求。

而面向对象正是解决以上这些问题的一种方法，可以对大量零散代码进行有效组织，从而使 PHP 具备大型 Web 项目开发的能力，同时还可以提高网站的易维护性和易读性。

专家点评

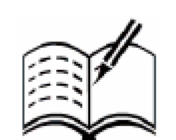
面向对象是 PHP 编程的核心技术，它相对于以往的面向过程编程来说，使得程序的设计更加合理，而且易于维护，所以在开发程序时，建议大家将面向对象的编程思想应用到项目开发中。

问题 176　构造方法和析构方法有什么不同？

问题阐述

程序中使用类时有两种特殊的方法，分别是构造方法和析构方法，它们有什么不同呢？





专家解答

构造方法和析构方法都具有与类相同的名称，下面分别对它们进行介绍。

1. 构造方法

构造方法是对象创建完成后第一个被对象自动调用的方法，它存在于每个声明的类中，是一个特殊的成员方法。如果在类中没有直接声明构造方法，那么类中会默认生成一个没有任何参数且内容为空的构造方法。

在 PHP 中，构造方法的声明有两种情况：第一种在 PHP5 以前的版本中，构造方法的名称必须与类名相同；第二种在 PHP5 的版本中，构造方法的方法名称必须是以两个下划线开始的"__construct()"。虽然在 PHP5 中构造方法的声明方法发生了变化，但是以前的方法仍然是可用的。

PHP5 中的这个变化是考虑到构造函数可以独立于类名，当类名发生变化时不需要修改相应的构造函数的名称。通过__construct()声明构造方法的语法如下：

```
function __construct([mixed args [,…]]){
        //方法体
}
```

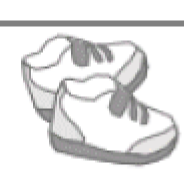
在 PHP 中，一个类只能声明一个构造方法。在构造方法中可以使用默认参数，实现其他面向对象的编程语言中构造方法重载的功能。如果在构造方法中没有传入参数，那么将使用默认参数为成员变量进行初始化。

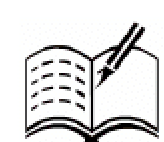
例如，使用__construct()声明一个与类名不同的构造方法。其代码如下：

```
<?php
/*
构造函数：当类被实例化后构造函数自动执行。所以如果用户希望在实例化的同时调用某个方法可以把此方法通过 this 关键字调用。
*/
    class mysql{                                                        //定义类名称
        var $localhost;                                                 //定义变量
        var $name;
        var $pwd;
        var $db;
        var $conn;
        public function __construct($localhost,$name,$pwd,$db){         //构造函数
            $this->localhost=$localhost;
            $this->name=$name;
            $this->pwd=$pwd;
            $this->db=$db;
            $this->connect();
        }
        //省略了部分代码
    }
    $msl=new mysql("127.0.0.1","root","111","db_database");         //实例化对象
    $msl->GetId();                                                  //对象句柄调用指定的方法
?>
```

2. 析构方法

析构方法的作用和构造方法正好相反，它是对象被销毁之前最后一个被对象自动调用的方





Note

法。该方法是 PHP5 中新添加的内容，实现在销毁一个对象之前执行一些特定的操作，诸如关闭文件、释放内存等。

析构方法的声明格式与构造方法类似，都是以两个下划线开头的“__destruct”，析构函数没有任何参数。其语法格式如下：

```
function __destruct(){
        //方法体，通常是完成一些在对象销毁前的清理任务
}
```

在 PHP 中有一种“垃圾回收”机制，可以自动清除不再使用的对象，从而释放内存。而析构方法就是在这个垃圾回收程序执行之前被调用的方法，在 PHP 中它属于类中的可选内容。

专家点评

通过上面的学习可以看出，构造函数和析构函数是完全不同的两个概念，构造函数用来初始化新对象，并为对象分配内存，而析构函数用来进行垃圾回收。

问题 177　为什么要封装？

问题阐述

面向对象编程中的一个重要特征就是封装，那么，为什么要实现封装呢？

专家解答

面向对象编程的特点之一是封装性，将类中的成员属性和方法结合成一个独立的相同单位，并尽可能隐藏对象的内容细节。其目的就是确保类以外的部分不能随意存取类的内部数据（成员属性和成员方法），从而有效避免外部错误对类内数据的影响。

此外，封装还可以解决数据存取的权限问题，可以使用封装将数据隐藏起来形成一个封闭的空间，然后可以设置哪些数据只能在这个空间中使用，哪些数据可以在空间外部使用。如果一个类中包含敏感数据，有些人可以访问，有些人不能访问，如果不对这些数据的访问加以限制，后果将会非常严重。所以在编写程序时，要对类的成员使用不同的访问修饰符，从而定义它们的访问级别。

类的封装可以通过关键字 public、private、protected、static 和 final 来实现。

应用示例

在本示例中通过 private（私有成员）定义私有变量，然后对变量进行修改与访问，并且直接调用私有变量，看会有什么情况发生。其代码如下：

```
<?php
    class Car{
        private $carName="奥迪系列";                //定义私有变量并赋值
        public function setName($carName){          //利用 set 方法为设置变量值
            $this->carName=$carName;
        }
        public function getName(){                  //利用 get 方法返回变量值
            return $this->carName;
        }
```



```
        }
        class SmallCar extends Car{                    //继承
        }
        $car=new SmallCar();                           //实例化子类对象
        $car->setName("Q7");                           //为子类变量赋值
        echo "正确操作私有变量<br>";
        echo $car->getName();                          //输出子类变量的值
        echo "<br>错误操作私有变量";
        echo Car::$carName;
?>
```



运行结果如图 8.2 所示。

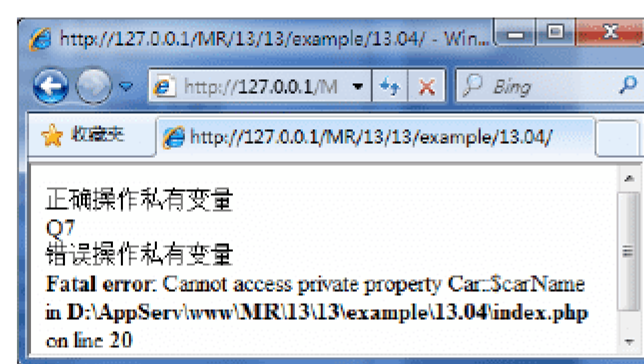


图 8.2　private 关键字

专家点评

对于成员方法，如果没有写关键字，那么默认就是 public。为了建立一个良好的编程习惯，建议读者在定义方法与变量时都加上关键字。

问题 178　如何继承一个类？

问题阐述

如何继承一个类？

专家解答

继承是 PHP5 面向对象程序设计的重要特性之一，类的继承就是指子类能够共享父类成员变量和成员方法的一种机制，这是类与类之间的一种关系的体现。通过类继承机制，可以利用已有的父类来派生新的子类，子类不仅可以拥有新定义的成员变量和成员方法，而且还同时拥有父类所有的成员变量和成员方法。

在 PHP 中，要实现类的继承可以使用关键字 extends，其语法格式如下：

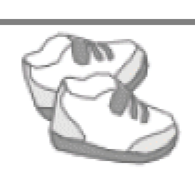
```
class Parent1 {                                  //定义父类 Parent1
    //父类的成员
}
class Child extends Parent1   {                  //定义子类 Child
    //子类的成员
}
```

通过上述继承关系即可在子类中使用 parent 关键字加“::”操作符调用父类的公有方法或属性，或者重写非 final 关键字修饰的方法。

应用示例

在本示例中，首先定义父类水果类，并在该类中定义获得水果颜色和水果形状的方法，然后定义水果类的子类苹果类，在苹果类中只定义构造方法用于实现对类进行初始化，最后通过调用父类的方法获得苹果的颜色和形状。其具体步骤如下：

（1）定义水果类 Fruit，在该类中定义颜色属性$color 和形状属性$shape，并通过构造方法





对这两个属性进行初始化，然后定义用于获得这两个属性的 getColor()方法和 getShape()方法。代码如下所示：

```
class Fruit {
                                                        //定义水果类
    private $color;                                     //颜色属性
    private $shape;                                     //形状属性
    public function __construct ($color, $shape)  {     //构造方法对水果类初始化
        $this->color = $color;
        $this->shape = $shape;
    }
    public function getColor () {                       //获得水果颜色
        return $this->color;
    }
    public function getShape () {                       //获得水果形状
        return $this->shape;
    }
}
```

（2）定义苹果类 Apple，使该类继承自水果类 Fruit，并定义苹果类的构造方法，在构造方法中使用 parent 关键字调用父类构造方法对父类进行初始化。代码如下所示：

```
class Apple extends Fruit    {                          //苹果类继承水果类
    public function __construct ($color, $shape) {      //构造方法对类进行初始化
        parent::__construct($color, $shape);
    }
}
```

（3）建立苹果颜色和形状录入表单。用户在表单中输入苹果的颜色和形状后单击“提交”按钮，将通过如下代码打印苹果的颜色和形状属性。

```
if(isset($_POST['color']) && $_POST['color']!=''){      //判断是否已经提交表单
    require 'Fruit.php';                                //包含水果类
    $apple = new Apple($_POST['color'], $_POST['shape']);   //对水果类进行实例化
    echo '<font color="blue">我看见的苹果是：'.$apple->getColor().','.$apple->getShape().'的</font>';
                                                        //打印苹果属性
}
```

上述代码首先判断用户是否已经提交表单，如果是，则首先使用 require 语句包含苹果类，然后使用 new 关键字对苹果类进行实例化，最后使用苹果类实例的对象调用父类中的 getColor()方法和 getShape()方法打印苹果的属性。

运行本示例，结果如图 8.3 所示。首先在表单中输入苹果的颜色和形状属性，然后单击“提交”按钮即可在页面中打印出苹果的颜色和形状属性。

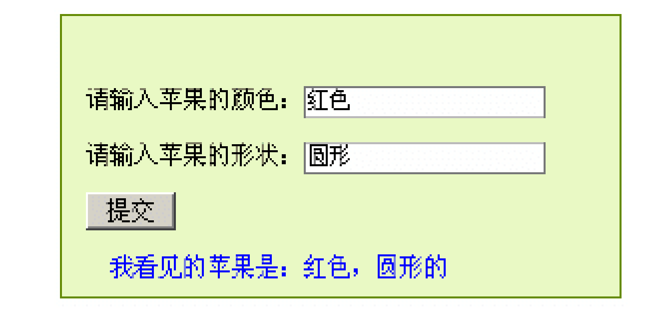
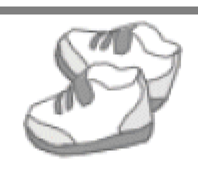


图 8.3　打印苹果颜色和形状属性

专家点评

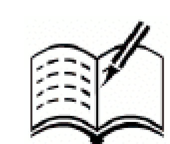
本示例中已经说明父类中使用 final 关键字所修饰的方法不能被重写，那么如何使一个类不能被其他类所继承呢？同样可以通过 final 关键字来修饰，如下所示：

```
final class Fruit    //使用 final 关键字修饰类{
}
```



上述的 Fruit 类被 final 关键字所修饰，这样该类就不能被其他类所继承。

问题 179　抽象类与接口的区别是什么？



问题阐述

抽象类与接口的区别是什么？

专家解答

1. 抽象类

抽象类是一种不能被实例化的类，只能作为其他类的父类来使用。抽象类使用 abstract 关键字来声明，其语法格式如下：

```
abstract class  抽象类名称{
    //抽象类的成员变量列表
    abstract function  成员方法 1(  参数  );                //定义抽象方法
    abstract function  成员方法 2(  参数  );                //定义成员方法
}
```

抽象类和普通类相似，包含成员变量、成员方法。两者的区别在于，抽象类至少要包含一个抽象方法。抽象方法没有方法体，其功能的实现只能在子类中完成，其也是使用 abstract 关键字来修饰。

注意：

在抽象方法后面要有分号“;”。

抽象类和抽象方法主要应用于复杂的层次关系中，这种层次关系要求每一个子类都包含并重写某些特定的方法。例如，中国的美食是多种多样的，有吉菜、鲁菜、川菜、粤菜等。每种菜系使用的都是煎、炒、烹、炸等手法，只是在具体的步骤上各有各的不同。如果把中国美食当作一个大类 Cate，下面的各大菜系就是 Cate 的子类，而煎、炒、烹、炸则是每个类中都有的方法。每个方法在子类中的实现都是不同的，在父类中无法规定。为了统一规范，不同子类的方法要有一个相同的方法名：decoct（煎）、stir_fry（炒）、cook（烹）、fry（炸）。

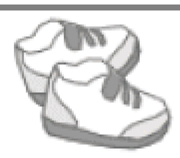
2. 接口

接口的作用是实现 PHP 的多重继承。接口类通过 interface 关键字来声明，接口中声明的方法必须是抽象方法，接口中不能声明变量，只能使用 const 关键字声明为常量的成员属性，并且接口中所有成员都必须具备 public 的访问权限。接口声明的语法格式如下：

```
interface  接口名称{                      //使用 interface 关键字声明接口
    //常量成员                            //接口中成员只能是常量
    //抽象方法;                           //成员方法必须是抽象方法
}
```

接口和抽象类相同都不能进行实例化的操作，也需要通过子类来实现。但是接口可以直接使用接口名称在接口外去获取常量成员的值。

例如，下面声明一个 One 接口，其代码如下：



```
interface One{                                        //声明接口
    const CONSTANT='CONSTANT value';                  //声明常量成员属性
    function FunOne();                                //声明抽象方法
}
```

接口之间也可以实现继承，同样需要使用 extends 关键字。

例如，下面声明一个 Two 接口，通过 extends 关键字继承 One。其代码如下：

```
interface Two extends One{                            //声明接口，并实现接口之间的继承
    function FunTwo();                                //声明抽象方法
}
```

抽象类和接口既有相同之处，也有类似的功能，但是它们之间还是存在很多区别的：

- ☑ 抽象类的操作通过继承关键字 extends 来实现，而接口的使用则通过 implements 关键字来实现。
- ☑ 抽象类中有数据成员，可以实现数据的封装，但是接口没有数据成员。
- ☑ 抽象类中可以有构造函数，但是接口没有构造函数。
- ☑ 抽象类的方法可以通过 private、protected 或 public 关键字来修饰，而接口中的方法只能使用 public 来修饰。
- ☑ 一个类只能继承于一个抽象类，而一个类可以同时实现多个接口。
- ☑ 抽象类中可以有成员方法的实现代码，而接口中不可以有成员方法的实现代码。

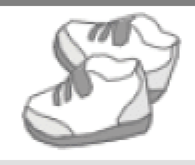
问题 180　如何封装一个项目工具类？

问题阐述

封装项目工具类是将项目中某些特殊的功能封装到一个类中，在项目需要使用这些特殊功能时，直接调用类中的方法即可。在这个类中，封装的方法包括：获取网站的相对路径、货币的格式化、HTML 标记的转义输出、中文字符串的截取、加载 FCKEditor 在线编辑器、输出 FCKEditor 编辑器的编辑结果以及关键字描红。

专家解答

```
/**************************************************************
 * @ 说明：Util 项目工具类，可进行货币格式化、HTML 转义、中文字符串截取及关键字描红等操作
 * @ 作者：明日科技
 * @ 学习社区：www.mrbccd.com
 * @ E-mail:mingrisoft@mingrisoft.com
 **************************************************************/
class Util{
    var $arrayIni;
    /**
     * 构造方法
     *
     * @return Util
     */
    function Util (){
        $this->arrayIni = parse_ini_file('config/lzhConfig.ini');
```





```
}
/*
 * @ 方法说明：
 *    网站相对地址
 */
function baseUrl (){
      $arrayIni = $this->arrayIni;
      return $arrayIni['baseUrl'];
}
/*
 * @ 方法说明：
 *    对货币进行格式化
 *
 * @ 参数说明：
 *    $money：金额
 */
function moneyFormat ($money){
      return str_replace(',', '', number_format($money, 2));
}
/*
 * @ 方法说明：
 *    对 html 标记进行转义输出
 *
 * @ 参数说明：
 *    $text：需转义的文本
 */
function unHtml ($text){
      $str = htmlspecialchars($text);
      $str = str_replace(chr(13), "<br>", $str);
      $str = nl2br($str);
      $str = str_replace(chr(32), " ", $str);
      return $str;
}
/*
 * @ 方法说明：
 *    对中文进行截取，防止出现乱码
 *
 * @ 参数说明：
 *    $str：要截取的字符串
 *    $start：开始截取的位置
 *    $len：截取的长度
 */
function msubstr ($str, $start, $len){
      $tmpstr=mb_substr($str,$start,$len,"utf-8");
      return $tmpstr;
}
/*
 * @ 方法说明：
 *    集成 FCKEditor
 *
 * @ 参数说明：
 *    $name：编辑器名称
```



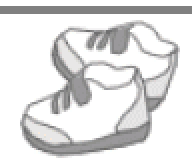

```
         *     $value：编辑器内容
         *     $width：编辑器宽度
         *     $height：编辑器高度
         */
        function editor ($name, $value, $width = '100%', $height = '200'){
            require_once 'library/fckeditor/fckeditor.php';
            $arrayIni = $this->arrayIni;
            $oFCKeditor = new FCKeditor($name);
            $oFCKeditor->BasePath = $arrayIni['baseUrl'] . '/library/fckeditor/';
            $oFCKeditor->Width = $width;
            $oFCKeditor->Height = $height;
            $oFCKeditor->ToolbarSet = 'Default';
            $oFCKeditor->Value = $value;
            $oFCKeditor->Create();
        }
        /*
         * @ 方法说明：
         *     对编辑器输出内容转义
         *
         * @ 参数说明：
         *     $text：输出内容
         */
        function editorUnHtml ($text){
            return stripslashes($text);
        }
        /*
         * @ 方法说明：
         *     对关键字进行描红
         *
         * @ 参数说明：
         *     $arrayKey：关键字
         *     $text：查询范围
         */
        function setRed ($arrayKey, $text){
            $tmpText = $this->unHtml($text);
            for ($i = 0; $i < count($arrayKey); $i ++) {
                if ($arrayKey[$i] != '') {
                    $tmpText = str_replace(trim($arrayKey[$i]), "<font color=\"#FF0000\">".$arrayKey
[$i] . "</font>", $tmpText);
                }
            }
            return $tmpText;
        }
    }
```

应用示例

在应用 Util 项目工具类时，首先进行类的实例化，然后通过实例化返回的对象调用类中的方法，实现对应的功能。类实例化的代码如下：

```
    $util=new Util();                                    //实例化项目工具类
```

通过 Util 类中的 baseUrl()方法获取网站中的相对地址。



```
<div style="width:930px; height:21px; background:url(<?php echo $util->baseUrl; ?> /img/title4.gif); border-top:1px solid #CCCCCC; padding-top:5px; text-align:left; padding-left:20px; font-size:13px">
```

通过 Util 类中的 moneyFormat()方法对货币进行格式化。

```
<?php echo $util->moneyFormat($res[$i]['m_price']);?>
```

通过 Util 类中的 unHtml()方法对 HTML 标记进行转义输出。

```
<?php echo $util->unHtml($res[$i]['area']);?>
```

通过 Util 类中的 msubstr()方法对字符串进行截取。

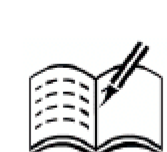
```
<?php echo $util->msubstr($res[$i]['name'],0,4);?>
```

通过 Util 类中的 editor()方法集成 FCKEditor 在线编辑器。

```
<?php echo $util->editor ('content', '') ;?>
```

通过 Util 类中的 setRed()方法对指定的关键字进行描红。此时对字符串中的“00”进行描红。

```
<?php echo $util->setRed ('00',$util->msubstr($res[$i]['addtime'],0,4));?>
```



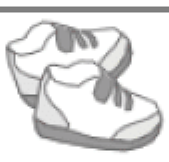
问题 181　如何将购物车中的功能封装到购物车类中？

问题阐述

购物车的主要功能是保留用户选择的商品信息。用户可以在购物车内设置选购商品的数量、显示选购商品的总金额，同时还可以执行清除选择的全部商品信息、重新选择商品信息等操作。为了提高开发的效率，如何将购物车中的各种功能都封装到 Cart 购物车类中，在需要各种功能时，只需调用类中对应的方法即可？

专家解答

```
/*******************************************************************
 * @ 说明：购物车类
 * @ 作者：明日科技
 * @ 学习社区：www.mrbccd.com
 * @ E-mail:mingrisoft@mingrisoft.com
 *******************************************************************/
class Cart{
    var $idStr; //商品 ID 组成的字符串，用字符@进行连接
    var $numStr; //商品数量组成的字符串，用字符@进行连接
    /*
     * @ 方法说明：
     *    构造方法
     *
     * @ 参数说明：
     *    $idStr：存储商品 ID 的字符串
     *    $numStr：存储商品数量的字符串
     */
    function Cart ($idStr, $numStr)
    {
        $this->idStr = $idStr; //为 ID 串赋初值
```

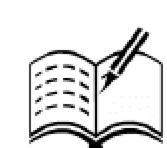




Note

```
        $this->numStr = $numStr; //为数量串赋初值
    }
    /*
     * @ 方法说明：
     *    将商品添加到购物车
     *
     * @ 参数说明：
     *    $goodsId：商品 ID
     *    $goodsNum：购买数量
     */
    function addCart ($goodsId, $goodsNum)
    {
        $arrayIds = explode('@', $this->idStr);
        $flag = 0;
        for ($i = 0; $i < count($arrayIds); $i ++) {
            if ($arrayIds[$i] == $goodsId) {
                $flag = 1;
            }
        }
        if ($flag == 0) {
            $this->idStr .= $goodsId . '@';
            $this->numStr .= $goodsNum . '@';
        } else {
            echo "<script>alert('该商品已经添加！');</script>";
        }
    }
    /*
     * @ 方法说明：
     *    将商品从购物车中移出
     *
     * @ 参数说明：
     *    $goodsId：商品 ID
     */
    function removeCart ($goodsId)
    {
        $arrayIds = explode('@', $this->idStr);
        $arrayNums = explode('@', $this->numStr);
        for ($i = 0; $i < count($arrayIds); $i ++) {
            if ($arrayIds[$i] == $goodsId) {
                unset($arrayIds[$i]);
                unset($arrayNums[$i]);
            }
        }
        $this->idStr = implode('@', $arrayIds);
        $this->numStr = implode('@', $arrayNums);
    }
    /*
     * @ 方法说明：
     *    更改购物车中的商品数量
     *
     * @ 参数说明：
     *    $goodsId：商品 ID
     *    $goodsNum：购买数量
```





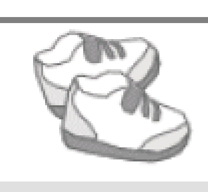
```
            */
            function changeNum ($goodsId, $goodsNum)
            {
                $arrayIds = explode('@', $this->idStr);
                $arrayNums = explode('@', $this->numStr);
                for ($i = 0; $i < count($arrayIds); $i ++) {
                    if ($arrayIds[$i] == $goodsId) {
                        $arrayNums[$i] = $goodsNum;
                    }
                }
                $this->idStr = implode('@', $arrayIds);
                $this->numStr = implode('@', $arrayNums);
            }
            /*
             * @ 方法说明：
             *    清空购物车
             */
            function setCartNull ()
            {
                $this->idStr = '';
                $this->numStr = '';
            }
            /*
             * @ 方法说明：
             *    获取购物车中商品 ID 字符串
             */
            function getIdStr ()
            {
                return $this->idStr;
            }
            /*
         * @ 方法说明：
         *    获取购物车中商品数量字符串
         */
            function getNumStr ()
            {
                return $this->numStr;
            }
        }
```

应用示例

明日科技书店作为一个电子商务类网站，购物车功能是必不可少的。下面笔者就以明日科技书店为背景，介绍如何应用购物车类实现购物车功能。其具体步骤如下：

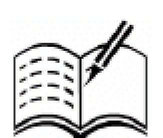
（1）购物车中商品的增、删、改等业务逻辑操作都是在 cart.php 文件中实现的，为了区分不同的商品购买者，可以应用$_SESSION 全局数组来保存商品 ID 串和商品数量串，所以在 cart.php 文件的代码前部应分别对保存商品 ID 串和商品数量串的$_SESSION 数组元素进行初始化设置。其具体实现代码如下：

```
    if(! isset($_SESSION['idStr'])||!isset($_SESSION['numStr'])){//判断是否设置商品 ID 的 SESSION 变量
        $_SESSION['idStr'] = '';                          //如果没有，则设置二者，并赋空值
        $_SESSION['numStr'] = '';
```



```
    }
    $cart = new Cart($_SESSION['idStr'], $_SESSION['numStr']);    //实例购物车类
```

上述代码首先判断用于保存商品 ID 串的$_SESSION['idStr']数组元素和保存商品数量串的$_SESSION['numStr']数组元素是否已经被设置，如果没有被设定，则说明当前用户是第一次将商品添加到购物车中，则将这两个数组元素的值设置为空，最后还需要对购物车类 Cart 进行实例化，以便继续对购物车中商品执行增、删、改操作。





（2）将商品添加到购物车是通过购物车类的 addCart()方法实现的，在讲解购物车类时已经介绍过相关方法的参数说明。该方法所包含的两个参数分别为购买商品的 ID 和购买商品的数量，将商品成功添加到购物车后，还需要使用购物车类的 getIdStr()方法和 getNumStr()方法分页获取经购物车类处理后的商品 ID 串和商品数量串，将获取结果分别赋给$_SESSION 全局数组中用于保存商品 ID 串和商品数量串的数组元素，从而实现购物车中的商品刷新效果。实现上述过程的代码如下：

```
if($t == 'add') {                                //判断当期操作类型
    $cart->addCart($_GET['bid'], 1);             //调用购物车类的 addCart()方法将商品添加到购物车中
    $_SESSION['idStr'] = $cart->getIdStr();          //获取商品 ID 串
    $_SESSION['numStr'] = $cart->getNumStr();        //获取商品数量串
}
```

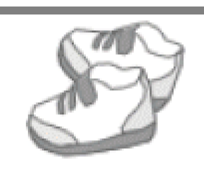
（3）在介绍图书详细信息页时，笔者已经介绍过如何将与某本图书相关的所有图书批量添加到购物车的实现过程，其原理是首先通过函数 explode()将保存组合图书 ID 的字符串用字符“@”分割，并将分割结果保存到数组中，这样该数组的每个元素为各本组合图书的 ID 号，然后通过 for 循环遍历该数组，在遍历过程中分别通过购物车类的 addCart()方法将图书添加到购物车中。上述过程的具体实现代码如下：

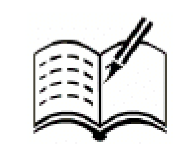
```
if($t == 'gAdd') {                                       //判断操作类型，如果为 gAdd 表示组合添加
    $arrayGIds = explode('@', $_GET['ids']);             //分割组合 ID 并将结果保存到数组中
    for($i=0; $i<count($arrayGIds); $i++) {              //遍历数组
        if($arrayGIds[$i]!='' && !in_array($arrayGIds[$i], explode('@', $_SESSION['idStr']))){
            $cart->addCart($arrayGIds[$i] ,1);           //将每本图书分别添加到购物车
        }
    }
    $_SESSION['idStr'] = $cart->getIdStr();              //重新为商品 ID 串的 SESSION 变量赋值
    $_SESSION['numStr'] = $cart->getNumStr();            //重新为商品数量串的 SESSION 变量赋值
}
```

（4）将指定的商品从购物车中移出是通过购物车类的 removeCart()方法实现的，该方法只包含要移出商品的ID参数。调用该方法后，再分别为保存商品ID串和商品数量串的$_SESSION全局数组重新赋值，从而实现刷新购物车的效果。该过程的实现代码如下：

```
if($t == 'remove')   {                        //判断当前操是否为删除操作
    $cart->removeCart($_GET['rbid']);         //调用购物车类的 removeCart()方法删除指定的图书
    $_SESSION['idStr'] = $cart->getIdStr();
    $_SESSION['numStr'] = $cart->getNumStr();
}
```

（5）在实现更改商品数量功能时，由于商品数量由购买者录入，所以在进行更新商品数量逻辑前，首先应判断购买者录入的数字是否合法，如果不合法，则给出提示并要求用户重新填写要更改的购买数量，成功通过验证后，就可以通过购物车类的 changNum()方法实现更改购





物车中商品的业务逻辑。该过程的实现代码如下：

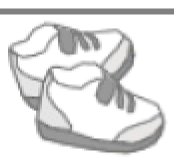
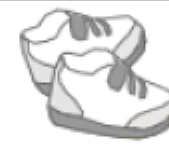
```
if($t == 'update') {                                    //如果$t 的值为 update，则进行更新操作
    $changenum = $_GET['num'];                          //获得要更新图书的数量
    if(!is_numeric($changenum) || intval($changenum)<=0 || floor($changenum)!=$changenum){
                                                        //判断数量是否为数字
        echo "<script>alert('购买图书的数量只能为正整数！');</script>";
    }else {
//调用购物车类的 changeNum()方法实现更改购物车中的图书数量
        $cart->changeNum($_GET['ubid'], $changenum);
        $_SESSION['idStr'] = $cart->getIdStr();
        $_SESSION['numStr'] = $cart->getNumStr();
    }
}
```

（6）清空购物车中商品是通过购物车类的 setCartNull()方法实现的，调用该方法后再重新为保存商品 ID 串和商品数量串的$_SESSION 全局数组赋值即可。其实现代码如下：

```
if($t == 'clearAll') {                         //如果$t 为 clearAll，则表示进行清空购物车操作
    $cart->setCartNull();                      //调用购物车类的 setCartNull()方法清空购物车
    $_SESSION['idStr'] = $cart->getIdStr();
    $_SESSION['numStr'] = $cart->getNumStr();
}
```

（7）完成购物车中商品的增、删、改操作后，还需要根据保存购物车商品数量串的$_SESSION['idStr']数组元素从数据库中查询出购物车中各本图书的信息，并对每本图书进行价格小计和购物车中所有图书的价格进行汇总。具体实现代码如下：

```
$arrayIds = explode('@', $_SESSION['idStr']);          //分割商品 ID 串为数组
$arrayNums = explode('@', $_SESSION['numStr']);        //分割商品数量串为数组
$arrayCarInfos = array();                              //保存购物车信息的数组
$totalPrice = 0;                                       //购物车图书总价格
for($i = 0; $i < count($arrayIds); $i++){
    $bookid = $arrayIds[$i];
    if($bookid != ''){
        $tmpArray = array();
        $bookinfo = $adminDB->executeSQL("select id, bookname ,oldprice, newprice from
tb_bookinfo where id='".$bookid."'", $connID);         //获得购物车每本图书的信息
        $tmpArray['id'] = $bookinfo[0]['id'];
        $tmpArray['bookname'] = $bookinfo[0]['bookname'];
        $tmpArray['oldprice'] = $bookinfo[0]['oldprice'];
        $tmpArray['newprice'] = $bookinfo[0]['newprice'];
        $tmpArray['num'] = $arrayNums[$i];
        $tmpArray['smallTotalPrice'] = $bookinfo[0]['newprice'] * $arrayNums[$i];
      $totalPrice += $tmpArray['smallTotalPrice'];     //计算图书总价
        array_push($arrayCarInfos, $tmpArray);         //将每本图书信息保存到数组中
    }
}
$smarty->assign('arrayCarInfos', $arrayCarInfos);
$smarty->assign('totalPrice', $totalPrice);
if(isset($_SESSION['unc']) && $_SESSION['unc']!='') {  //判断用户是否登录
    $smarty->assign('isLogin', 'T');
}else{
```

```
        $smarty->assign('isLogin', 'F');
        if(isset($_SESSION['toUrl'])){
            unset($_SESSION['toUrl']);
        }
        $_SESSION['toUrl'] = 'getbuyuserinfo.html';
    }
```

创建的购物车如图 8.4 所示。



图 8.4　购物车

问题 182　如何制作柱形图绘制类?

问题阐述

如何制作柱形图绘制类?

专家解答

绘制柱形图应用的是 Jpgraph 类库中的 BarPlot 类，调用 SetFillGradient()方法完成颜色、形状的设置。

（1）SetFillGradient()方法，设置柱形图的颜色、形状。其语法如下：

```
SetFillGradient($aFromColor, $aToColor, $aStyle)
```

参数说明：

❶ $aFromColor：默认颜色。

❷ $aToColor：转换的颜色。

❸ $aStyle：设置的样式。

（2）SetFont()方法统计图标题、坐标轴等文字样式。其语法如下：

```
SetFont($family, [$style,] [$size])
```

参数说明：

❶ $family：指定文字的字体。

❷ $style：指定文字的样式。

❸ $size：指定文字的大小，默认为 10。

（3）SetMargin()方法设置图像、标题、坐标轴上文字与边框的距离。其语法如下：



```
SetMargin($left,$right,$top,$bottom)
```

参数指定其与左右、上下边框的距离。

或者：

```
SetMargin($data)
```

参数$data 同样指定与边框的距离。

应用示例

在本示例中运用 Jpgraph 生成柱形图，对公司编程词典上半年的销量进行统计，其运行结果如图 8.5 所示。

（1）创建 index.php 文件，设置网页的编码格式，并通过 include()语句导入 Jpgraph 类库。

（2）应用 Jpgraph 类库中的方法完成柱形图的创建，其关键代码如下：

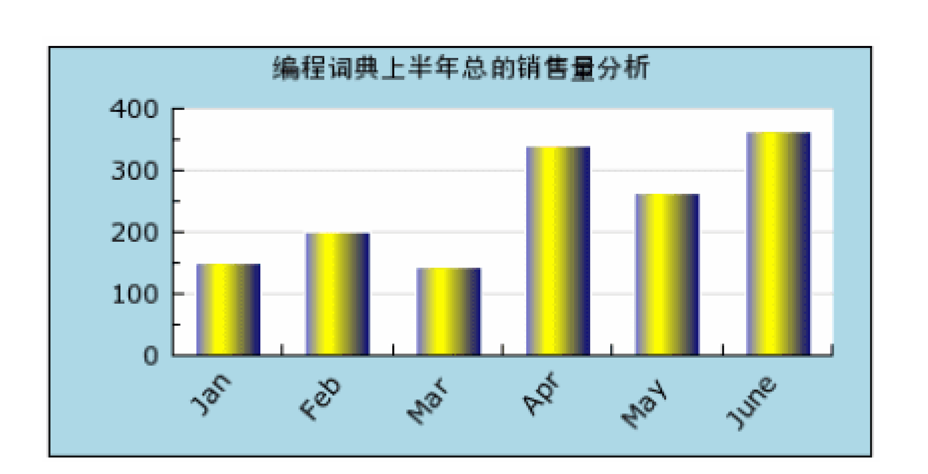


图 8.5　编程词典上半年销量展示

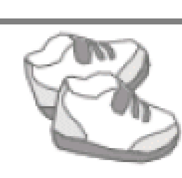


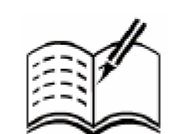
```
<?php
header ( "Content-type: text/html; charset=UTF-8" );                //设置文件编码格式
include ("../src/jpgraph.php");
include ("../src/jpgraph_bar.php");
$datay=array(150,201,145,340,265,365);
$datax=array("Jan","Feb","Mar","Apr","May","June");
$graph = new Graph(400,200,"auto");                                 //创建图像
$graph->img->SetMargin(60,20,30,50);                                //设置图像边框间距
$graph->SetScale("textlin");                                        //定义坐标刻度类型
$graph->SetMarginColor("lightblue");                                //定义图像颜色
$graph->title->Set(iconv("utf-8","gb2312",'编程词典上半年总的销售量分析'));  //定义标题
$graph->title->SetFont(FF_SIMSUN, FS_BOLD);                         //设置标题字体
$graph->xaxis->SetFont(FF_VERDANA,FS_NORMAL,10);                    //设置 X 轴的字体
$graph->yaxis->SetFont(FF_VERDANA,FS_NORMAL,10);                    //设置 Y 轴的字体
$graph->xaxis->SetTickLabels($datax);                               //设置 X 轴输出的数据
$graph->xaxis->SetLabelAngle(50);                                   //设置输出文字大小
$bplot = new BarPlot($datay);                                       //实例化图像创建类
$bplot->SetWidth(0.6);                                              //设置柱形图的输出大小
$bplot->SetFillGradient("navy","#FFFF00",GRAD_LEFT_REFLECTION);
                                                                    //设置图像的类型和填充颜色
$bplot->SetColor("white");                                          //设置图像边框颜色
$graph->Add($bplot);                                                //添加数据
$graph->Stroke();                                                   //生成图像
?>
```

问题 183　如何制作折线图绘制类？

问题阐述

如何制作折线图绘制类？





Note

专家解答

绘制折线图应用的是 Jpgraph 类库中的 LinePlot 类。输出数据应用的是 value()对象中的 Show()方法，折线图中标记的添加应用的是 mark 对象的 SetType()方法。

（1）使用 LinePlot 对象绘制曲线。

通过 Jpgraph 类库中的 LinePlot 类创建曲线。该类的语法如下：

```
$linePlot = new LinePlot($data)                    //创建折线图
```

参数说明：

$data：数值型数组，指定统计数据。

（2）输出数据应用的是 value()对象中的 Show()方法，数据的格式化应用的是 SetFormat()方法。

```
SetFormat($aFormat, $aNegFormat)
```

参数说明：

❶ $aFormat：设置格式化的样式。

❷ $aNegFormat：设置 negative 的值。

（3）折线图中标记的添加应用的是图像中的 mark 对象的 SetType()方法，其参数值设置标记的类型。其语法如下：

```
SetType($aType, $aFileName, $aScale)
```

参数说明：

❶ $aType：设置标记的类型。

❷ $aFileName：插入图像的名称或国家的名称。

❸ $aScale：设置插入标记的大小。

下面列举了一些 SetType()方法中第一个参数可选的类型：MARK_SQUARE、MARK_UTRIANGLE、MARK_DTRIANGLE、MARK_DIAMOND、MARK_CIRCLE。

应用示例

在本示例中运用 Jpgraph 生成折线图，对公司 2010 年的销售额进行分析，其运行结果如图 8.6 所示。

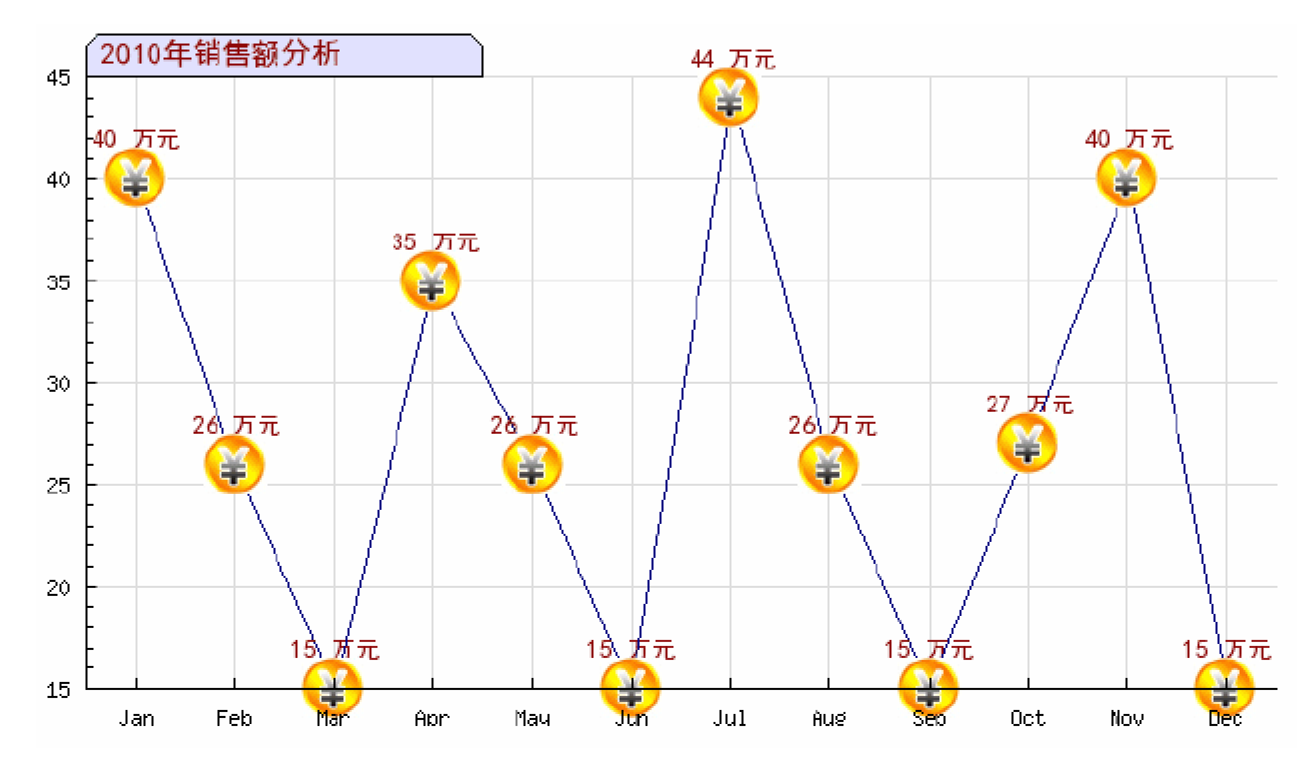
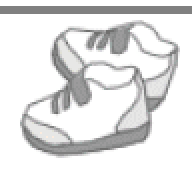


图 8.6　2010 年公司的销售额

折线图分析 2010 年公司销售额走势的关键步骤如下：



（1）创建 index.php 文件，设置网页的编码格式，并通过 include()语句导入 Jpgraph 类库。

（2）应用 Jpgraph 类库中的方法创建一个折线图，对 2010 年公司的销售额进行分析。其代码如下：

```
<?php
header ( "Content-type: text/html; charset=UTF-8" );                      //设置文件编码格式
include ("../src/jpgraph.php");
include ("../src/jpgraph_line.php");
include ("../src/jpgraph_scatter.php");
$datay1 = array(40,26,15,35,26,15,44,26,15,27,40,15);
$graph = new Graph(650,375);                                               //定义图像大小
$graph->SetMarginColor('white');                                           //背景颜色
$graph->SetScale("textlin");                                               //定义刻度值类型
$graph->SetFrame(false);
$graph->SetMargin(30,5,35,20);                                             //设置边距
$graph->tabtitle->Set(iconv("utf-8","gb2312",'2010 年销售额分析') ); //输出标题
$graph->tabtitle->SetFont(FF_SIMSUN, FS_BOLD,12);                          //设置标题字体
$graph->tabtitle->SetColor('darkred','#E1E1FF');                           //设置标题颜色
$graph->xgrid->Show();                                                     //设置阴影
$graph->xaxis->SetTickLabels($gDateLocale->GetShortMonth());               //定义 X 轴上的数据
$p1 = new LinePlot($datay1);                                               //创建图像
$p1->SetColor("navy");                                                     //设置图像颜色
$p1->mark->SetType(MARK_IMG,'saab_95.jpg',0.5);                            //设置标签的样式，使用图片
$p1->value->SetFormat('%d '.iconv("utf-8","gb2312",'万元'));               //格式化数据
$p1->value->Show();                                                        //输出阴影
$p1->value->SetColor('darkred');                                           //定义颜色
$p1->value->SetFont(FF_SIMSUN, FS_BOLD,10);                                //设置字体
$p1->value->SetMargin(14);                                                 //设置位置、字体大小
$p1->SetCenter();
$graph->Add($p1);                                                          //添加数据
$graph->Stroke();                                                          //生成图像
?>
```

问题 184　如何制作饼形图绘制类？

问题阐述

如何制作饼形图绘制类？

专家解答

绘制 3D 饼形图应用的是 PiePlot3D 类，继承 PiePlot 类。

通过 PiePlot3D 类中的 Explode()方法对饼形图进行分割。Explode()方法语法如下：

```
Explode($aExplodeArr)
```

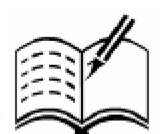
将饼形图分割成多个模块，参数$aExplodeArr 是一个数组，根据数组元素个数定义分割成几块，根据数组元素值定义分割后模块与原图像的距离。

通过饼形图对象调用 SetTheme()方法设置饼形图的颜色。SetTheme()方法的语法如下：



```
PiePlot ::
SetTheme($aTheme)
```

参数$aTheme 指定主题的名称。其支持的主题包括："earth"、"pastel"、"sand"、"water"。通过饼形图对象调用 SetTheme()方法设置饼形图的颜色。SetTheme()方法的语法如下：



```
PiePlot ::
SetTheme($aTheme)
```

参数$aTheme 指定主题的名称。其支持的主题包括："earth"、"pastel"、"sand"、"water"。

应用示例

在本示例中运用 Jpgraph 生成饼形图，对公司 2010 年图书销量进行分析，其运行结果如图 8.7 所示。

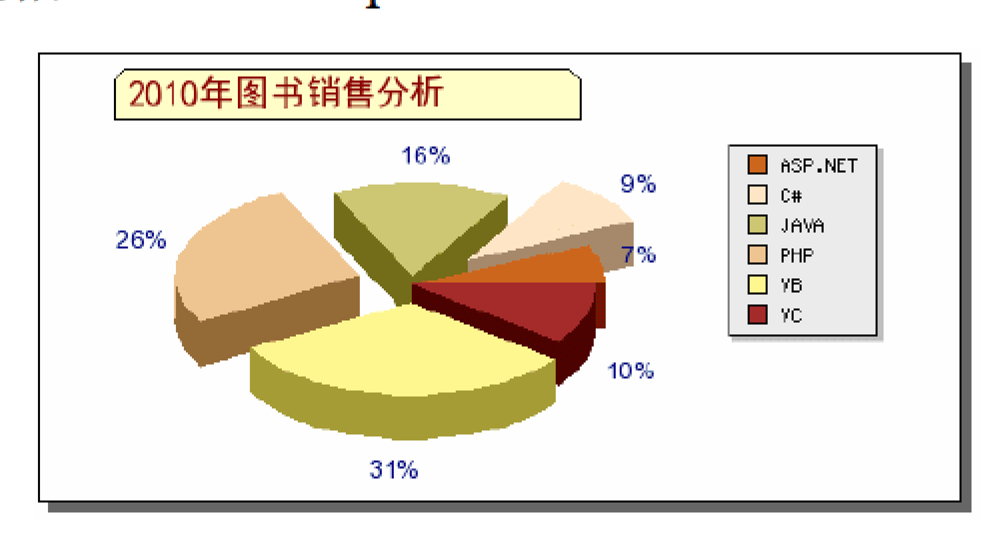


图 8.7　2010 年图书销量分析

饼形图分析 2010 年图书销量的关键步骤如下：

（1）创建 index.php 文件，设置网页的编码格式，通过 include()语句导入 Jpgraph 类库。

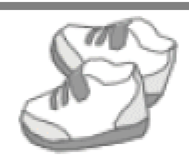
（2）应用 Jpgraph 类库中的方法创建饼形图，对 2010 年图书销量进行分析。其代码如下：

```
<?php
header ( "Content-type: text/html; charset=UTF-8" );                //设置文件编码格式
include ("../src/jpgraph.php");
include ("../src/jpgraph_pie.php");
include ("../src/jpgraph_pie3d.php");
$data = array(20,27,45,75,90,30);
$graph = new PieGraph(500,245,"auto");                              //创建图像
$graph->SetShadow();                                                //创建图像阴影
$graph->tabtitle->Set(iconv("utf-8","gb2312",'2010 年图书销售分析') );  //输出标题
$graph->tabtitle->SetFont(FF_SIMSUN, FS_BOLD,14);                   //设置标题字体
$graph->title->SetColor("darkblue");                                //定义标题颜色
$graph->legend->Pos(0.1,0.2);                                       //控制注释文字的位置
$p1 = new PiePlot3d($data);                                         //创建图像
$p1->SetTheme("sand");                                              //控制图像颜色
$p1->SetCenter(0.4);                                                //设置图像位置
$p1->SetSize(0.4);                                                  //设置图像大小
$p1->SetHeight(20);                                                 //设置饼形图高度
$p1->SetAngle(45);                                                  //设置图像倾斜角度
$p1->Explode(array(5,40,10,30,20));                                 //控制饼形图的分割
$p1->value->SetFont(FF_ARIAL,FS_NORMAL,10);                         //设置字体
$p1->SetLegends(array("ASP.NET","C#","JAVA","PHP","VB","VC"));
$graph->Add($p1);                                                   //添加数据
$graph->Stroke();                                                   //生成图像
?>
```

问题 185　如何封装验证码类？

问题阐述

验证码功能在注册、登录、发布信息和评论中经常会用到，为了减少代码的冗余以及提高





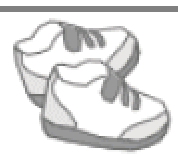
代码的重用率，可以将该功能封装到类中，在需要时直接调用类中方法即可实现验证码功能。那么如何封装验证码类呢？

专家解答

1. 类的封装

首先，定义验证码类的名称是 ValidateCode。该类的语法如下：

```
/*********************************************
 * @ 说明：验证码类
 * @ 作者：明日科技
 * @ 学习社区：www.mrbccd.com
 * @ E-mail:mingrisoft@mingrisoft.com
 ***********************************************/
class ValidateCode{
    //宽度
    private $_width;
    //长度
    private $_height;
    //验证码
    private $_codeStr;
    //字体类型
    private $_fontType;
    //图像句柄
    private $_img;
     /**
     * 构造方法
     * @param string $width
     * @param string $height
     * @param string $codeStr
     * @param int $fontType
     */
    public function __construct ($width, $height, $codeStr = '0000', $fontType = 0){
        $this->_width = $width;
        $this->_height = $height;
        $this->_codeStr = substr($codeStr, 0, 4);
        $this->_fontType = $fontType;
    }
    /**
     * 获取颜色
     *
     * @param 颜色范围最小值 $x
     * @param 颜色范围最大值 $y
     * @return imagecolorallocate()
     */
    private function _getColor ($x, $y) {
        $r = mt_rand($x, $y);
        $g = mt_rand($x, $y);
        $b = mt_rand($x, $y);
        return imagecolorallocate($this->_img, $r, $g, $b);
    }
```




Note

```
	/**
	 * 初始图像
	 *
	 */
	private function _init (){
		$this->_img = imagecreate($this->_width, $this->_height);
	}
	/**
	 * 创建验证码
	 *
	 */
	private function _build () {
		imagefill($this->_img, 0, 0, $this->_getColor(150, 250));
		imagerectangle($this->_img, 0, 0, $this->_width - 1, $this->_height - 1, $this->_getColor(50, 150));
		if ($this->_fontType == 0) {
			$fontFileName = 'ARIALBI.TTF';
		} else {
			$fontFileName = 'ARIALN.TTF';
		}
		for ($i = 0; $i < strlen($this->_codeStr); $i ++) {
			imagettftext($this->_img, mt_rand(12, 24), 0, ($this->_width) / 4 * $i, mt_rand(20, 
$this->_height - 5), $this->_getColor(10, 180), 'font/'. $fontFileName, substr($this->_codeStr, $i, 1));
		}
		for ($i = 0; $i < 15; $i ++) {
			imageline($this->_img, mt_rand(0, $this->_width), mt_rand(0, $this->_height), 
mt_rand(0, $this->_width), mt_rand(0, $this->_height), $this->_getColor(110, 210));
		}
	}
	/**
	 * 显示图像
	 *
	 */
	public function show (){
		header('content-type:image/png');
		$this->_init();
		$this->_build();
		imagepng($this->_img);
	}
}
```

2. 应用示例

在本示例中编写一个用户登录功能，调用验证码类生成验证码，通过 jQuery 在客户端完成对用户登录信息的验证。其具体步骤如下：

（1）封装 ValidateCode 验证码类，存储于 ValidateCode.php 文件中。

（2）创建 index.php 文件，设计用户登录界面，添加用户登录的表单元素。

（3）创建 yzm.php 文件，生成验证码，并且在 index.php 文件中以图像形式输出。其关键代码如下：

```
<?php
session_start();                                              //初始化 SESSION 变量
```



```
include("ValidateCode.php");                              //载入验证码生成类
$codeStr = "";                                            //定义验证码变量
$codeArray = array('0' , '1' , '2' , '3' , '4' , '5' , '6' , '7' , '8' , '9' , 'A' , 'B' , 'C' , 'D' , 'E' , 'F' , 'G' , 'H' , 'I' , 'J' ,
'K' , 'L' , 'M' , 'N' , 'O' , 'P' , 'Q' , 'R' , 'S' , 'T' , 'U' , 'V' , 'W' , 'X' , 'Y' , 'Z');
for ($i = 0; $i < 4; $i ++) {
    $codeStr .= $codeArray[mt_rand(0, 35)];               //生成随机验证码数据
}
$_SESSION['Vcode']=$codeStr;                              //将验证码数据赋给 SESSION 变量
$validateCode = new ValidateCode(80, 30, $codeStr, 0);    //实例化验证码类
$validateCode->show();                                    //生成验证码
?>
```

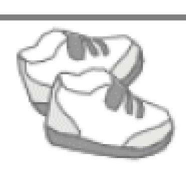


（4）创建 JavaScript 脚本，编写 changeVcode()方法，实现验证码的切换。其代码如下：

```
function changeVcode(){
    $("#vcodeImg").attr("src", "yzm.php?rand="+Math.random()+"");
}
```

（5）编写 JavaScript 脚本，通过 jQuery 技术完成对表单中提交数据的验证。其中要特别注意的是调用的 check_code.php 文件，在该文件中对用户提交的验证码进行验证。JavaScript 脚本的代码如下：

```
//表单验证
function chkInputLogin(flag){
        $.get("check_code.php?vcode="+encodeURI($("#vcode").val()), null, function(chkVcodeReturnData){
                var k1=false;
                var k2=false;
                var k3=false;
                //用户昵称
                if(flag == 1 || flag == -1){
                    if($.trim($("#netname").val())==""){
                        $("#chkNetname").html("      请输入登录昵称");
                        $("#netname").css("border", "1px solid #FF0000");
                        $("#chkNetname").css("display", "block");
                    }else{
                        $("#chkNetname").html("");
                        $("#netname").css("border", "1px solid #777777");
                        $("#chkNetname").css("display", "none");
                        k1=true;
                    }
                }
                //登录密码
                if(flag == 2 || flag == -1){
                    if($.trim($("#password").val())==""){
                        $("#chkPassword").html("      请输入登录密码");
                        $("#password").css("border", "1px solid #FF0000");
                        $("#chkPassword").css("display", "block");
                    }else{
                        $("#chkPassword").html("");
                        $("#password").css("border", "1px solid #777777");
                        $("#chkPassword").css("display", "none");
                        k2=true;
```





Note

```
                }
            }
            //验证码
            if(flag == 3 || flag == -1){
                if($.trim($("#vcode").val())==""){
                    $("#chkVcode").html("    请输入验证码");
                    $("#vcode").css("border", "1px solid #FF0000");
                    $("#chkVcode").css("display", "block");
                }else if(chkVcodeReturnData == "N"){
                    $("#chkVcode").html("    验证码输入有误");
                    $("#vcode").css("border", "1px solid #FF0000");
                    $("#chkVcode").css("display", "block");
                }else{
                    $("#chkVcode").html("");
                    $("#vcode").css("border", "1px solid #777777");
                    $("#chkVcode").css("display", "none");
                    k3=true;
                }
            }
            //提交表单
            if(flag == -1 && k1 && k2 && k3 ){
                $("#form_login").submit();
            }
        });
    }
```

运行结果如图 8.8 所示。

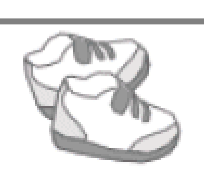


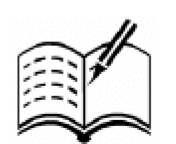
图 8.8　用户登录时的验证码



数据库抽象层

- ⏭ 问题 186　什么是 PDO?
- ⏭ 问题 187　如何安装 PDO?
- ⏭ 问题 188　如何通过 PDO 访问数据库?
- ⏭ 问题 189　如何通过 PDO 读取数据表中的数据?
- ⏭ 问题 190　如何解决 SQL 查询中的错误?
- ⏭ 问题 191　如何向数据库中添加或修改数据?
- ⏭ 问题 192　如何保护 Web 站点免受 SQL 注入攻击?
- ⏭ 问题 193　如何获得受影响的行数?
- ⏭ 问题 194　如何进行事务处理?
- ⏭ 问题 195　如何利用 PDO 使用存储过程?
- ⏭ 问题 196　如何利用 PDO 连接 Oracle 11g?
- ⏭ 问题 197　如何封装 PDO 数据库连接类?
- ⏭ 问题 198　如何封装 PDO 数据库操作类?
- ⏭ 问题 199　如何封装 PDO 分页类?
- ⏭ 问题 200　如何封装 ADODB 连接数据库类?
- ⏭ 问题 201　ADODB 操作结果集的方法有哪些?
- ⏭ 问题 202　ADODB 中如何控制结果集的存取方法?
- ⏭ 问题 203　如何一次连接两个数据库?
- ⏭ 问题 204　如何在 ADODB 中生成 HTML 表格?
- ⏭ 问题 205　如何在 ADODB 中生成下拉列表框?
- ⏭ 问题 206　一个小巧的分页函数——ADODB_Pager()
- ⏭ 问题 207　如何通过事务处理机制保证数据库操作的完整性?
- ⏭ 问题 208　ADODB 如何输出系统中执行的 SQL 语句?
- ⏭ 问题 209　ADODB 如何应用 ErrorMsg 返回错误信息?
- ⏭ 问题 210　ADODB 如何返回所有错误信息?
- ⏭ 问题 211　如何将错误信息存储于日志文件中?
- ⏭ 问题 212　如何封装 ADODB 操作数据库类?
- ⏭ 问题 213　如何封装 ADODB 分页类?





Note

问题 186　什么是 PDO?

问题阐述

什么是 PDO？PDO 有什么特点？

专家解答

PDO 是 PHP Date Object（PHP 数据对象）的简称，它是与 PHP 5.1 版本一起发行的，目前支持的数据库包括 Firebird、FreeTDS、Interbase、MySQL、MSSQL Server、ODBC、Oracle、Postgre SQL、SQLite 和 Sybase。有了 PDO，用户不必再使用 mysql_*、oci_*及 mssql_*函数，也不必再为它们封装数据库操作类，只需要使用 PDO 接口中的方法就可以对数据库进行操作。在选择不同的数据库时，只需修改 PDO 的 DSN（数据源名称）即可。

在 PHP 6 中将默认使用 PDO 连接数据库，所有非 PDO 扩展将会在 PHP 6 中被移除。该扩展提供 PHP 内置类 PDO 来对数据库进行访问，不同数据库使用相同的方法名解决数据库连接不统一的问题。

PDO 是一个“数据库访问抽象层”，其作用是统一各种数据库的访问接口，与 MySQL 和 MSSQL 函数库相比，PDO 让跨数据库的使用更具有亲和力；与 ADODB 和 MDB2 相比，PDO 更高效。

PDO 将通过一种轻型、清晰、方便的函数，统一各种不同 RDBMS 库的共有特性，实现 PHP 脚本最大程度的抽象性和兼容性。

PDO 吸取现有数据库扩展成功和失败的经验教训，利用 PHP 5 的最新特性，可以轻松地与各种数据库进行交互。

PDO 扩展是模块化的，使其能够在运行时为用户的数据库后端加载驱动程序，而不必重新编译或安装整个 PHP 程序。例如，PDO_MySQL 扩展会替代 PDO 扩展实现 MySQL 数据库 API。另外，还有一些用于 Oracle、PostgreSQL、ODBC 和 Firebird 的驱动程序，更多的驱动程序尚在开发。

问题 187　如何安装 PDO?

问题阐述

如何安装 PDO？

专家解答

PDO 是与 PHP 5.1 一起发行的，默认包含在 PHP 5.1 中。由于 PDO 需要 PHP 5 核心面向对象特性的支持，因此其无法在 PHP 5.0 之前的版本中使用。

在默认情况下，PDO 在 PHP 5.2 中为开启状态，但是要启用对某个数据库驱动程序的支持，仍需要进行相应的配置操作。

在 Linux 环境下，要使用 MySQL 数据库，可以在 configure 命令中添加如下选项：

```
--with-pdo-mysql=/path/to/mysql/installation
```

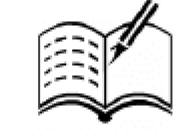


在 Windows 环境下，PDO 在 php.ini 文件中进行配置，如图 9.1 所示。

要启用 PDO，首先必须加载“extension=php_pdo.dll”，如果要想其支持某个具体的数据库，那么还要加载对应的数据库选项。例如，要支持 MySQL 数据库，则需要加载“extension=php_ pdo_mysql.dll”选项。



图 9.1　Windows 环境下配置 PDO



专家点评

在完成数据库的加载后还要保存 php.ini 文件，并且重新启动 Apache 服务器，修改才能够生效。

问题 188　如何通过 PDO 访问数据库？

问题阐述

如何通过 PDO 访问数据库？

专家解答

在 PDO 中，要建立与数据库的连接需要实例化 PDO 的构造函数。PDO 构造函数的语法如下：

```
__construct(string $dsn[,string $username[,string $password[,array $driver_options]]])
```

参数说明：

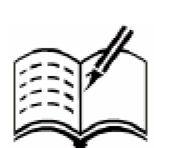
❶ dsn：数据源名，包括主机名端口号和数据库名称。
❷ username：连接数据库的用户名。
❸ password：连接数据库的密码。
❹ driver_options：连接数据库的其他选项。

1. 连接 MySQL 数据库

下面的代码展现了如何连接到一个本地主机上的 MySQL 数据库。

```
<?php
    header("Content-Type:text/html;charset=utf-8");        //设置页面的编码格式
    $dbms='mysql';                                          //数据库类型
    $dbName='db_database';                                  //使用的数据库名称
    $user='root';                                           //使用的数据库用户名
    $pwd='111';                                             //使用的数据库密码
    $host='localhost';                                      //使用的主机名称
    $dsn="$dbms:host=$host;dbname=$dbName";
    try {                                                   //捕获异常
        $pdo=new PDO($dsn,$user,$pwd);                      //实例化对象
        echo "PDO 连接 MySQL 成功";
    } catch (Exception $e) {
        echo $e->getMessage()."<br>";
    }
?>
```





2. 连接 SQL Server 数据库

使用以下代码连接到本地主机上的 SQL Server 数据库。

```
<?php
    header("Content-Type:text/html;charset=utf-8")        //设置页面的编码风格
    $host='PC-201006101638';                              //设置主机名称
    $user='sa';                                           //设置用户名
    $pwd='';                                              //设置密码
    $dbName='db_database';                                //设置所要连接的数据库
    $dbms='mssql';
    $dsn="mssql:host=$host;dbname=$dbName";
    try {                                                 //利用 try…catch 捕捉异常并实现连接
        $pdo = new PDO($dsn,$user,$pwd);
        echo "连接 SQL SERVER 成功";
    } catch (Exception $e) {
        die("ERROR!".$e->getMessage()."<br>");
    }
?>
```

3. 连接 Oracle 数据库

使用以下代码连接到本地主机上的 Oracle 数据库。

```
<?php
$dbms='oci';                                  //数据库类型，对于开发者来说，使用不同的数据
                                                库，只要修改它即可，不用记住那么多的函数
$dbName='192.168.1.59:1521/oralcles';         //使用的数据库
$user='system';                               //数据库连接用户名
$pass='mrsoft';                               //对应的密码
$dsn="$dbms:dbname=$dbName";
try {
    $pdo = new PDO($dsn, $user, $pass);       //初始化一个 PDO 对象，就是创建了数据库连接
                                                对象$pdo
    echo "连接成功<br/>";
    $pdo = null;
} catch (PDOException $e) {
    die ("Error!: " . $e->getMessage() . "<br/>");
}
?>
```

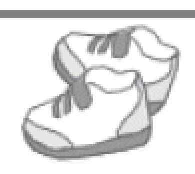
说明：

通过PDO连接Oracle数据库，首先要在本机上安装Oracle数据库的客户端。然后，去掉php.ini文件中 extension=php_pdo.dll 和 extension=php_pdo_oci.dll 前面的分号，加载 PDO 模块。最后，将 Oracle 客户端的 oraociei10.dll、oci.dll 和 orannzsbb10.dll 复制到 Apache 的 bin 文件夹下。

专家点评

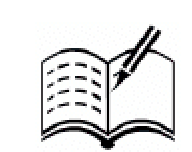
在使用不同的数据库时，必须明确数据库服务器是完全独立于 PHP 的实体，即数据库服务器可能与 Web 服务器不在同一台计算机上，此时要通过 PDO 连接数据库时，就需要修改 DSN 中的主机名称。

DSN 是 Data Source Name（数据源名称）的首字母缩写，其提供连接数据库需要的信息。



PDO 的 DSN 包括 3 部分：PDO 驱动名称（如 mysql、sqlite 或 pgsql）、冒号和驱动特定的语法。每种数据库都有其特定的驱动语法。

由于数据库服务器只在特定的端口上监听连接请求。每种数据库服务器具有一个默认的端口号（MySQL 是 3306），但是数据库管理员可以对端口号进行修改，因此有可能 PHP 找不到数据库的端口，此时就可以在 DSN 中包含端口号。



另外，由于一个数据库服务器中可能拥有多个数据库，所以在通过 DSN 连接数据库时，通常都包括数据库名称，这样可以确保连接的是用户想要的数据库，而不是其他人的数据库。

问题 189　如何通过 PDO 读取数据表中的数据?

问题阐述

既然通过 PDO 可以实现与数据库的连接，那么如何通过 PDO 读取数据表中的数据呢?

专家解答

PDO 为用户提供了很多与数据库交互的方法。其中最常用的是 exec()、query()、prepare()和 execute()方法。

（1）exec()方法返回执行后受影响的行数，其语法如下：

```
int PDO::exec ( string statement )
```

参数 statement 是要执行的 SQL 语句。该方法返回执行查询时受影响的行数，通常用于 INSERT、DELETE 和 UPDATE 语句中。

（2）query()方法通常用于返回执行查询后的结果集。其语法如下：

```
PDOStatement PDO::query ( string statement )
```

参数 statement 是要执行的 SQL 语句，它返回的是一个 PDOStatement 对象。

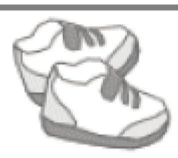
（3）预处理语句包含 prepare()和 execute()两个方法。首先，通过 prepare()方法做查询的准备工作，然后，通过 execute()方法执行查询，并且还可以通过 bindParam()方法来绑定参数提供给 execute()方法。其语法如下：

```
PDOStatement PDO::prepare ( string statement [, array driver_options] )
bool PDOStatement::execute ( [array input_parameters] )
```

1. 使用 query()方法执行查询语句

通过 query()方法执行 SQL 语句，循环输出查询结果。其关键代码如下：

```
<?php
$dbms='mysql';     //数据库类型，对于开发者来说，使用不同的数据库，只要修改它即可，不用记住那么多的函数
$host='localhost';                    //数据库主机名
$dbName='db_database';                //使用的数据库
$user='root';                         //数据库连接用户名
$pass='111';                          //对应的密码
$dsn="$dbms:host=$host;dbname=$dbName";
try {
    $pdo = new PDO($dsn, $user, $pass); //初始化一个 PDO 对象，就是创建了数据库连接对象$pdo
```






```
        $query="select * from tb_pdo_mysql";        //定义 SQL 语句
    $result=$pdo->query($query);                    //执行查询语句，并返回结果集
    foreach($result as $items){
        ?>
            <tr>
              <td height="22" align="center" valign="middle"><?php echo $items['id'];?></td>
              <td align="center" valign="middle"><?php echo $items['pdo_type'];?></td>
              <td align="center" valign="middle"><?php echo $items['database_name'];?></td>
              <td align="center" valign="middle"><?php echo $items['dates'];?></td>
            </tr>
            <?php
            }
            } catch (PDOException $e) {
        die ("Error!: " . $e->getMessage() . "<br/>");
    }
    ?>
```

运行结果如图 9.2 所示。

2. 使用 prepare()和 execute()方法执行查询语句

使用 prepare()和 execute()方法通常被认为是处理数据库查询的最理想的方法。在本示例中，首先以 SQL 语句为参数调用 PDO->prepare()方法，返回一个 PDOStatement 对象，然后调用 execute()方法执行查询语句，返回查询结果集，最后通过 while 循环重复调用 PDOStatement-> fetch()方法输出查询结果集。其关键代码如下：



ID	PDO	数据库	时间
1	pdo	mysql	2010-09-18 10:19:17
2	pdo	Oracle	2010-09-18 10:19:17
4	pdo	Access	2010-10-10 00:00:00
22	pdo	SQL	2010-10-09 10:59:30
23	pdo	db2	2010-10-16 10:59:41
24	pdo	mysql	2010-08-21 00:00:00
25	pdo	MySQL	2010-10-10 10:10:10
26	pdo	MySQL	2010-10-10 10:10:10

图 9.2　使用 PDO 的 query()方法产生的输出

```
    <?php
    $dbms='mysql';                          //数据库类型，对于开发者来说，使用不同的数据库，只
                                              要修改它即可，不用记住那么多的函数
    $host='localhost';                      //数据库主机名
    $dbName='db_database';                  //使用的数据库
    $user='root';                           //数据库连接用户名
    $pass='111';                            //对应的密码
    $dsn="$dbms:host=$host;dbname=$dbName";
    try {
    $pdo = new PDO($dsn, $user, $pass);     //初始化一个PDO对象，就是创建了数据库连接对象$pdo
        $query="select * from tb_pdo_mysql";                //定义 SQL 语句
        $result=$pdo->prepare($query);                      //准备查询语句
        $result->execute();                                 //执行查询语句，并返回结果集
        while($res=$result->fetch(PDO::FETCH_ASSOC)){  //循环输出查询结果集，并且设置结果
                                                              集为关联索引
        ?>
            <tr>
              <td height="22" align="center" valign="middle"><?php echo $res['id'];?></td>
              <td align="center" valign="middle"><?php echo $res['pdo_type'];?></td>
              <td align="center" valign="middle"><?php echo $res['database_name'];?></td>
              <td align="center" valign="middle"><?php echo $res['dates'];?></td>
            </tr>
    <?php
```



```
            }
                } catch (PDOException $e) {
            die ("Error!: " . $e->getMessage() . "<br/>");
        }
                ?>
```

Note

专家点评

在选择 SQL 语句的执行方法时，如果只执行一次查询，那么 PDO->query()是较好的选择。虽然它无法自动转义发送给它的任何数据，但它在遍历 SELECT 语句的结果集方面的确非常方便，然而在使用这个方法时应十分小心。如果没有在结果集中获取到所有数据，那么下次调用 PDO->query()方法时可能会失败。

如果要多次使用 SQL 语句，那么最好的方法是使用 prepare()和 execute()方法——这是优先选择的方案。使用 prepare()和 execute()方法与使用 query()方法比较起来有很多优势。首先，它通过自动转义任何提供给它的参数帮助防止 SQL 注入攻击。其次，多次使用（例如，执行多次数据库插入或更新）的预编译语句占用较少的资源，运行速度要比重复调用 query()方法快很多。另外，当用 prepare()和 execute()方法时，还可以选择很多种返回数据格式。例如，在示例中使用 fetch()方法时，定义结果集采用关联索引（PDO::FETCH_ASSOC）的形式。

问题 190　如何解决 SQL 查询中的错误？

问题阐述

错误不可避免，它们困扰所有的人，经常由于无法控制的情况——数据添加失败、数据更新失败或者不能正确删除数据等造成。那么，该如何捕获 SQL 语句中的错误，并及时加以解决呢？

专家解答

PDO 为捕获错误提供了如下 3 种解决方案。

☑　使用默认模式——PDO::ERRMODE_SILENT

在默认模式中设置 PDOStatement 对象的 errorCode 属性，但不进行其他任何操作。

☑　使用警告模式——PDO::ERRMODE_WARNING

警告模式会产生一个 PHP 警告，并设置 errorCode 属性。如果设置的是警告模式，那么除非明确地检查错误代码，否则程序将继续按照其方式运行。

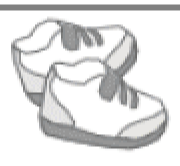
☑　使用异常模式——PDO::ERRMODE_EXCEPTION

异常模式会创建一个 PDOException，并设置 errorCode 属性。它可以将执行代码封装到一个 try{…}catch{…}语句块中。未捕获的异常将会导致脚本中断，并显示堆栈跟踪让用户了解是哪里出现的问题。

1. 采用默认模式捕获 SQL 语句中的错误

通过 prepare()和 execute()方法向数据库中添加数据，设置 PDOStatement 对象的 errorCode 属性，手动检测代码中的错误。主要代码如下：

```
<?php
if($_POST['Submit']=="提交" && $_POST['pdo']!=""){
    $dbms='mysql';                                //数据库类型，对于开发者来说，使用不同的数据
```




Note

```
                                        库，只要修改它即可，不用记住那么多的函数
        $host='localhost';              //数据库主机名
        $dbName='db_database';          //使用的数据库
        $user='root';                   //数据库连接用户名
        $pass='111';                    //对应的密码
        $dsn="$dbms:host=$host;dbname=$dbName";
        $pdo = new PDO($dsn, $user, $pass);   //初始化一个 PDO 对象，就是创建了数据库连接对
                                              象$pdo
        $query="insert into tb_pdo_mysqls(pdo_type,database_name,dates)values ('".$_POST['pdo']."',
'".$_POST['databases']."','".$_POST['dates']."')";
        $result=$pdo->prepare($query);
        $result->execute();
        $code=$result->errorCode();
        if(empty($code)){
            echo "数据添加成功！ ";
        }else{
            echo '数据库错误：<br/>';
            echo 'SQL Query:'.$query;
            echo  '<pre>';
            var_dump($result->errorInfo());
            echo '</pre>';
        }
    }
        ?>
```

在本示例中，在定义 INSERT 添加语句时使用了错误的数据表名称 tb_pdo_mysqls（正确名称是 tb_pdo_mysql），导致输出如图 9.3 所示的结果。

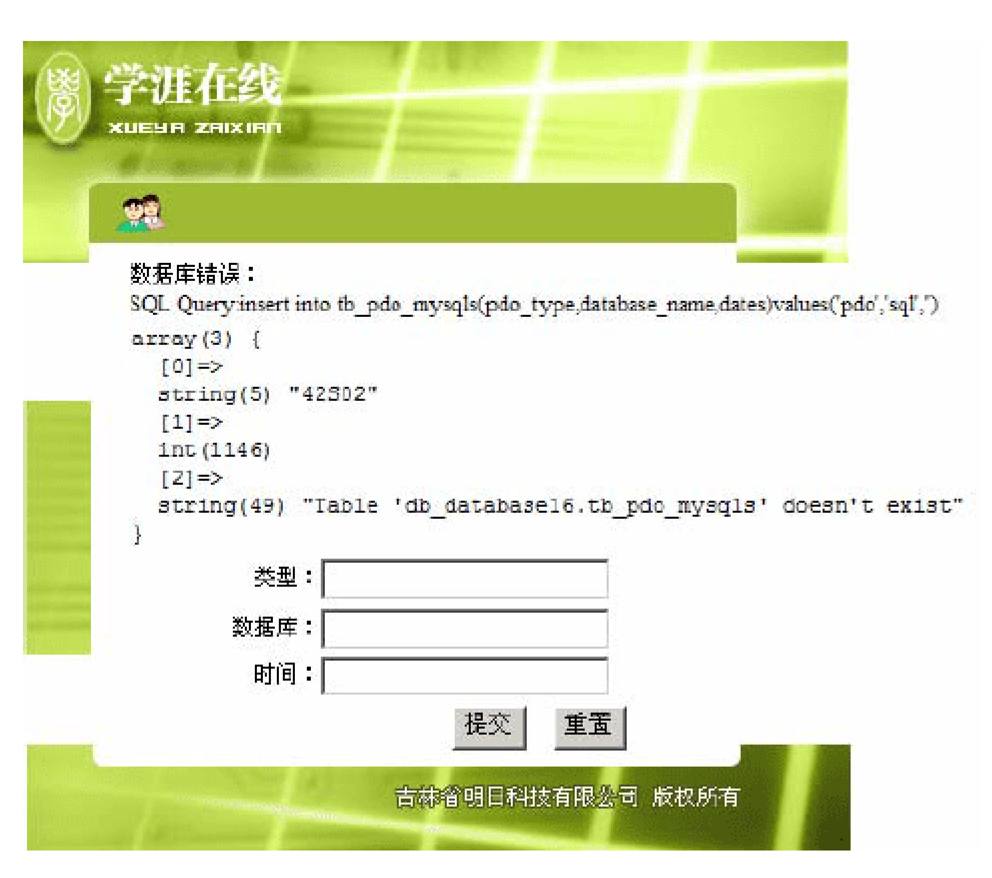


图 9.3　在默认模式中捕获 SQL 中的错误

2. 采用警告模式捕获 SQL 语句中的错误

设置警告模式通过 prepare()和 execute()方法读取数据库中的数据，并且通过 while 语句和 fetch()方法完成数据的循环输出，体会在设置成警告模式后执行错误的 SQL 语句。其关键代码如下：

```
    <?php
    $dbms='mysql';                      //数据库类型，对于开发者来说，使用不同的数据
                                          库，只要修改它即可，不用记住那么多的函数
    $host='localhost';                  //数据库主机名
```



```
$dbName='db_database';                          //使用的数据库
$user='root';                                   //数据库连接用户名
$pass='111';                                    //对应的密码
$dsn="$dbms:host=$host;dbname=$dbName";
try {
    $pdo = new PDO($dsn, $user, $pass);         //初始化一个 PDO 对象，就是创建了数据库连
                                                  接对象$pdo
    $pdo->setAttribute(PDO::ATTR_ERRMODE,PDO::ERRMODE_WARNING); //设置为警告模式
    $query="select * from tb_pdo_mysqls";       //定义 SQL 语句
    $result=$pdo->prepare($query);              //准备查询语句
    $result->execute();                         //执行查询语句，并返回结果集
    while($res=$result->fetch(PDO::FETCH_ASSOC)){  //while 循环输出查询结果集，并且设置
                                                     结果集为关联索引
    ?>
          <tr>
            <td height="22" align="center" valign="middle"><?php echo $res['id'];?></td>
            <td align="center" valign="middle"><?php echo $res['pdo_type'];?></td>
            <td align="center" valign="middle"><?php echo $res['database_name'];?></td>
            <td align="center" valign="middle"><?php echo $res['dates'];?></td>
          </tr>
<?php
    }
          } catch (PDOException $e) {
    die ("Error!: " . $e->getMessage() . "<br/>");
}
  ?>
```

运行结果如图 9.4 所示。

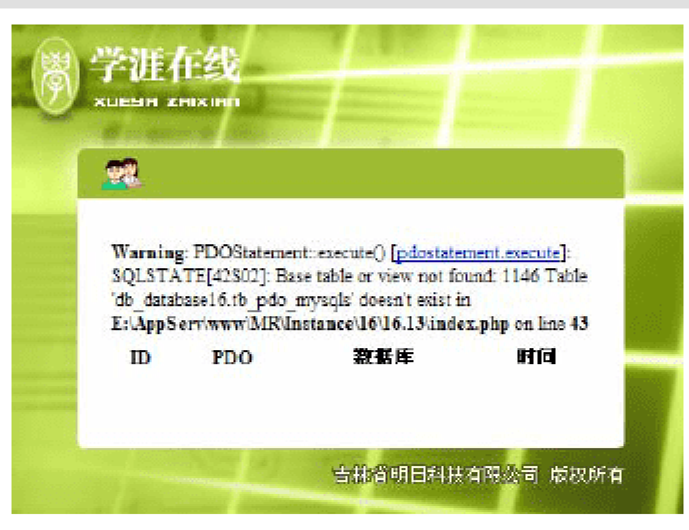


图 9.4　设置警告模式后捕获的 SQL 语句错误

3. 采用异常模式捕获 SQL 语句中的错误

在执行数据库中数据的删除操作时设置为异常模式，并且编写一个错误的 SQL 语句（操作错误的数据表 tb_pdo_mysqls），体会异常模式与警告模式和默认模式的区别。其关键代码如下：

```
<?php
header ( "Content-type: text/html; charset=utf-8" );   //设置文件编码格式
if($_GET['conn_id']!=""){
    $dbms='mysql';                              //数据库类型，对于开发者来说，使用不同的
                                                  数据库，只要修改它即可，不用记住那么多
                                                  的函数
    $host='localhost';                          //数据库主机名
    $dbName='db_database';                      //使用的数据库
    $user='root';                               //数据库连接用户名
    $pass='111';                                //对应的密码
    $dsn="$dbms:host=$host;dbname=$dbName";
    try {
        $pdo = new PDO($dsn, $user, $pass);     //初始化一个 PDO 对象，就是创建了数据库
                                                  连接对象$pdo
        $pdo->setAttribute(PDO::ATTR_ERRMODE,PDO::ERRMODE_EXCEPTION);
```





```
            $query="delete from tb_pdo_mysqls where Id=:id";
            $result=$pdo->prepare($query);                    //预准备语句
            $result->bindParam(':id',$_GET['conn_id']);       //绑定更新的数据
            $result->execute();
        } catch (PDOException $e) {
            echo 'PDO Exception Caught.';
            echo 'Error with the database:<br/>';
            echo   'SQL Query: '.$query;
            echo '<pre>';
            echo "Error: " . $e->getMessage(). "<br/>";
            echo "Code: " . $e->getCode(). "<br/>";
            echo "File: " . $e->getFile(). "<br/>";
            echo "Line: " . $e->getLine(). "<br/>";
            echo "Trace: " . $e->getTraceAsString(). "<br/>";
            echo '</pre>';
        }
    }
    ?>
```

在设置为异常模式后，执行错误的 SQL 语句返回的结果如图 9.5 所示。

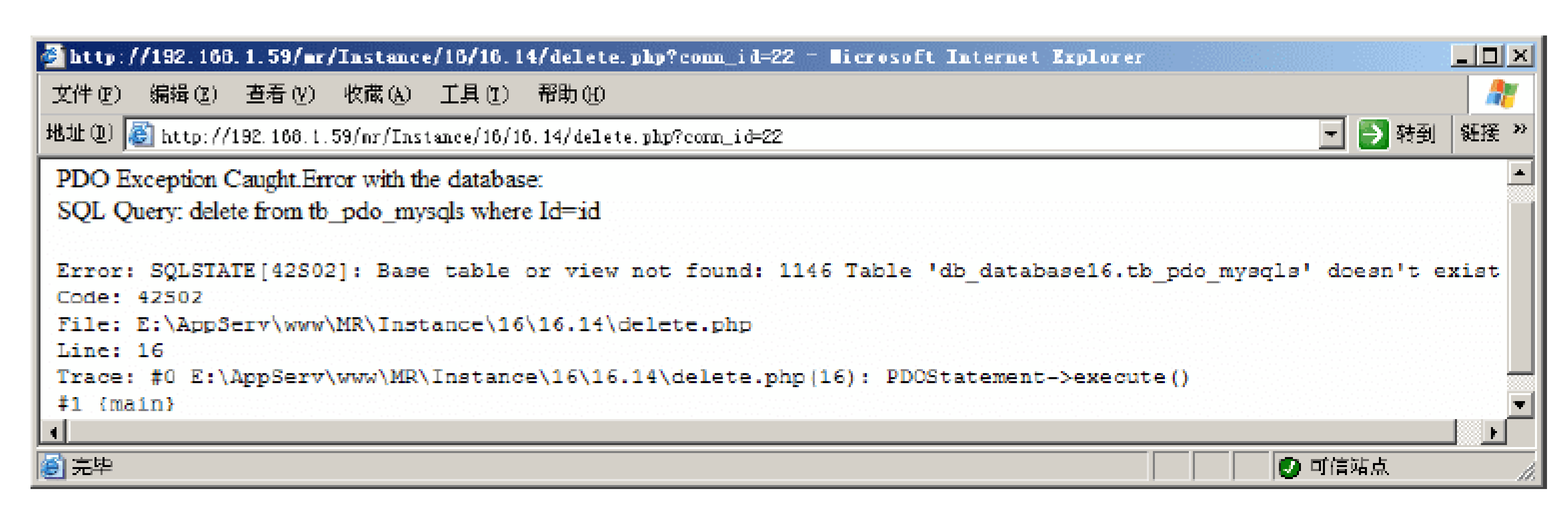


图 9.5　异常模式捕获的 SQL 语句错误信息

专家点评

由于异常模式符合错误处理的面向对象风格（捕获和处理不同类型的异常），而且易于使用，因此大多数人会选择利用 PHP 更强大的面向对象模型，并使用异常模式处理错误。

无论选择哪种错误处理方法，返回 SQL 查询本身的文本都是一个很好的想法，这将使用户能够确切了解查询是否存在问题，而且有助于进行错误调试。

问题 191　如何向数据库中添加或修改数据？

问题阐述

如何向数据库中添加或修改数据？

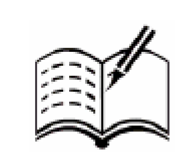
专家解答

可以使用 SQL 中的 INSERT 语句向数据库中添加数据，使用 UPDATE 语句修改数据库中已有的数据。然后通过 query()方法或 prepare()和 execute()方法执行添加和更新语句。

应用示例

通过 prepare()和 execute()方法向数据库中添加数据，其具体实现步骤如下：





首先，设计网页页面，创建 form 表单将要添加的数据提交到本页。然后，定义数据库连接的参数，实例化 PDO 构造函数，完成与数据库的连接。接着，定义 INSERT 添加语句，通过$_POST[]方法获取表单中提交的数据。最后，通过 prepare()和 execute()方法向数据库中添加数据。其代码如下：

```
<?php
if($_POST['Submit']=="提交" && $_POST['pdo']!=""){
    $dbms='mysql';                        //数据库类型，对于开发者来说，使用不同的
                                           数据库，只要修改它即可，不用记住那么多
                                           的函数
    $host='localhost';                    //数据库主机名
    $dbName='db_database';                //使用的数据库
    $user='root';                         //数据库连接用户名
    $pass='111';                          //对应的密码
    $dsn="$dbms:host=$host;dbname=$dbName";
    try {
        $pdo = new PDO($dsn, $user, $pass);   //初始化一个 PDO 对象，就是创建了数据库连
                                               接对象$pdo
        $query="insert into tb_pdo_mysql(pdo_type,database_name,dates)values('".$_POST['pdo']."',
'".$_POST['databases']."','".$_POST['dates']."')";
        $result=$pdo->prepare($query);
        if($result->execute()){
            echo "数据添加成功！ ";
        }else{
            echo "数据添加失败！ ";
        }
    } catch (PDOException $e) {
        die ("Error!: " . $e->getMessage() . "<br/>");
    }
}
?>
```

专家点评

在实际的应用程序中，部分查询输入可能会来自于用户生成的内容。由于正在使用 prepare()和 execute()方法，因此不必担心 SQL 注入会攻击这个查询：所有变量都将被自动转义。

问题 192　如何保护 Web 站点免受 SQL 注入攻击？

问题阐述

SQL 注入攻击是指攻击者利用站点中合法用户的输入机制发送 SQL 代码到数据库执行。避免 SQL 注入攻击的黄金法则是：在来自外部源的所有数据到达数据库之前将其转义。该法则不仅适用于 INSERT 和 UPDATE 语句，也适用于 SELECT 查询。

在脚本的查询中使用预编译语句几乎可以消除所有的 SQL 注入攻击问题，但如果选择使用 query()方法，将不具备这种保护——必须对添加到查询中的用户输入手工转义。其代码如下：

```
<?php
    $dbms='mysql';
```





```
            $dbName='db_database';
            $user='root';
            $pwd='111';
            $host='localhost';
            $dsn="$dbms:host=$host;dbname=$dbName";
            //$name='mr';
            $name="'or Name LIKE'%";
            $query="select * from tb_zc where user='". $name ."'";        //SQL 语句
                    try {
                        $pdo=new PDO($dsn,$user,$pwd);
                        $result=$pdo->query($query);                 //输出结果集中的数据
                        foreach ( $result as $row){                   //输出结果集中的数据
    ?>
        <tr>
          <td bgcolor="#FFFFFF"><div align="center"><?php echo    $row['user'];?>;</div></td>
        </tr>
        <?php }
                        } catch (Exception $e) {
                            echo "ERROR!!".$e->getMessage()."<br>";
                            }
    ?>
```

在这个示例中，假定 SQL 语句中的$name 变量来自于用户提交的表单。通常用户会提交像“mr”这种内容，这将会生成下面的 SQL 语句：

```
select * from tb_zc where user='mr'
```

这在脚本中不会造成任何问题。然而，一个“聪明”的攻击者可能会输入“'OR Name LIKE'%”，从而得到下面的 SQL 语句：

```
select * from tb_zc where user=''OR Name LIKE'%'
```

这种输入会为攻击者打开整个表。如果是非常敏感的数据，将被完全暴露在外。那么采用什么方法才能解决这个问题呢？

专家解答

要想解决这个问题，可以使用 PDO 中的 quote()方法转义传递给 SQL 字符串的数据。只需要将 SQL 代码修改成：

```
$sql="select * from tb_zc where user='". $pdo->quote($name) ."'";
```

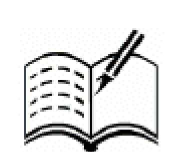
记住需要引用 SQL 查询中使用的每块数据，这个没有快捷途径，除非考虑使用 prepare()和 execute()方法。

专家点评

如果正在使用 PDO->query()方法，那么总需要使用 quote()方法引用用户的输入，记住是“总需要”。

如果选择使用 prepare()和 execute()方法，那么不必引用那些需要绑定到预编译 SQL 的值（例如，将被插入到数据库的值），驱动会为用户做所有的事。然而，有时可能无法将一个变量绑定到预编译 SQL 语句，在这种情况下，如果正在创建一个动态 SQL 语句，那么需要引用所有用到的无法被绑定的值（如 GROUP BY 子句或 ORDER BY 或表名称）。





问题 193 如何获得受影响的行数？

问题阐述

在做进一步动作前就能够计算出一个查询返回或影响的行数通常很有用。希望分页显示结果或生成统计信息时，这种功能特别方便。那么获取受影响行数的功能是如何实现的呢？

专家解答

1. 通过 SQL 聚合函数 COUNT 统计查询结果记录数

COUNT 返回一个查询或部分查询的行数，通常和 DISTINCT 一起使用。SQL 的聚合函数 COUNT 被多种数据库广泛支持。

2. 通过 PDOStatement->rowCount()方法返回受影响行数

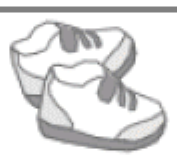
PDOStatement->rowCount()方法获取到有多少行被 UPDATE、INSERT 或 DELETE 查询所影响。rowCount()方法在典型的 PHP 应用程序中并不常见，但它可以统计出在调用 PDOStatement->execute 后，有多少行受到影响。

3. 应用示例

通过 SQL 聚合函数 COUNT 统计查询返回的记录数。其具体代码如下：

```
<?php
if(isset($_POST['Submit']) && $_POST['Submit']=="查询"){
$dsn = 'mysql:host=localhost;dbname=db_database;';
$user = 'root';
$password = '111';
$id =$_POST['nr'];
try{
   $dbh = new PDO($dsn, $user, $password);
   $dbh->setAttribute(PDO::ATTR_ERRMODE,
         PDO::ERRMODE_EXCEPTION);
   $sql = 'SELECT COUNT(*) FROM tb_pdo_mysql WHERE database_name=:id';
   $stmt = $dbh->prepare($sql);
   $stmt->bindParam(':id', $id, PDO::PARAM_STR);
   $result = $stmt->execute();
   echo 'There are ', $stmt->fetchColumn(), ' rows returned.';
}
catch (PDOException $e){
   echo 'PDO Exception Caught.   ';
   echo 'Error with the database: <br />';
   echo 'SQL Query: ', $sql;
   echo 'Error: ' . $e->getMessage();
}
}
?>
```

运行本示例，在文本框中输入要查询的数据库名称，单击“查询”按钮，即可显示出所查询到的结果行数。运行结果如图 9.6 所示。




Note



图 9.6　获取查询返回的行数

专家点评

在使用 UPDATE 和 DELETE 语句时，一定要随时确定添加了 WHERE 子句。如果未添加 WHERE 子句，则可能会更新数据库中的所有列或者删除表中的所有数据，无论是哪一种情况都不是用户希望看到的。

问题 194　如何进行事务处理？

问题阐述

如果需要将一组 SQL 查询作为一个操作来运行，为了保证数据的完整性，需要进行事务处理，几乎所有数据库都以这样或那样的形式提供事务支持，那么如何使用 PDO 进行事务处理呢？

专家解答

在 PDO 中实现事务处理的功能，其应用的方法如下：

☑　开启事务——beginTransaction()方法

beginTransaction()方法将关闭自动提交（autocommit）模式，直到事务提交或回滚以后才恢复。

☑　提交事务——commit()方法

commit()方法完成事务的提交操作，如果成功则返回 True，否则返回 False。

☑　事务回滚——rollback()方法

rollback()方法执行事务的回滚操作。

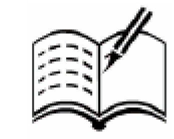
应用示例

通过 prepare()和 execute()方法向数据库中添加数据，并且通过事务处理机制确保数据能够正确地添加到数据中。其具体步骤如下：

创建 index.php 文件。首先，定义数据库连接的参数，创建 try{}catch{}语句，在 try{}语句中实例化 PDO 构造函数，完成与数据库的连接，并且通过 beginTransaction()方法开启事务。然后，定义 INSERT 添加语句，通过$_POST[]方法获取表单中提交的数据，通过 prepare()和 execute()方法向数据库中添加数据，并且通过 commit()方法完成事务的提交操作。最后，在 catch{}语句中返回错误信息，并且通过 rollBack()执行事务的回滚操作。其代码如下：

```
<?php
if($_POST['Submit']=="提交" && $_POST['pdo']!=""){
    $dbms='mysql';                          //数据库类型，对于开发者来说，使用不同的数据
                                             库，只要修改它即可，不用记住那么多的函数
```





```
                $host='localhost';                    //数据库主机名
                $dbName='db_database';                //使用的数据库
                $user='root';                         //数据库连接用户名
                $pass='111';                          //对应的密码
                $dsn="$dbms:host=$host;dbname=$dbName";
                try {
                    $pdo = new PDO($dsn, $user, $pass);   //初始化一个 PDO 对象，就是创建了数据库连
                                                           接对象$pdo
                    $pdo->beginTransaction();         //开启事务
                    $query="insert into tb_pdo_mysql(pdo_type,database_name,dates)values('".$_POST['pdo']."',
'".$_POST['databases']."','".$_POST['dates']."')";
                    $result=$pdo->prepare($query);
                    if($result->execute()){
                        echo "数据添加成功！ ";
                    }else{
                        echo "数据添加失败！ ";
                    }
                    $pdo->commit();                   //执行事务的提交操作
                } catch (PDOException $e) {
                      die ("Error!: " . $e->getMessage() . "<br/>");
                    $pdo->rollBack();                 //执行事务的回滚
                }
            }
            ?>
```

运行结果如图 9.7 所示。



图 9.7　数据添加中应用事务处理机制

专家点评

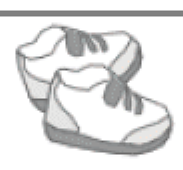
在 MySQL 中，事务由单独单元的一个或多个 SQL 语句组成。在这个单元中，每个 MySQL 语句是相互依赖的，而整个单独单元作为一个不可分割的整体，如果单元中一旦某条 SQL 语句执行失败或产生错误，整个单元将会回滚，所有受到影响的数据将返回到事务开始以前的状态；如果单元中的所有 SQL 语句均执行成功，则事务被顺利执行。但不是所有的数据库都支持事务，因此，默认情况下，PDO 运行在自动提交模式下，在这种情况下，每个查询被视为它自己的事务进行处理。如果数据库不支持事务，那么查询执行时也没有事务。如果数据库支持事务，那么可以不使用自动提交模式，而是使用手动开始和停止事务。

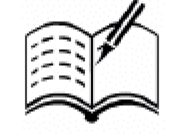
在上面的示例中，PDO-> beginTransaction()和$pdo->commit()方法在 try 块中被调用。Catch 块使用$pdo->rollBack()方法在出现问题时回滚数据库。

问题 195　如何利用 PDO 使用存储过程？

问题阐述

许多数据库支持存储过程——运行在数据库上的脚本，通常使用数据库特定的 SQL 语言





编写。存储过程允许在更接近于数据的位置操作数据，从而减少了带宽的使用。它们使数据独立于脚本逻辑，允许使用不同语言的多个系统以相同的方式访问数据，从而节省了花费在编码和调试上的宝贵时间。另外，存储过程使用预定的执行计划提高了查询速度，并且能够阻止与数据的直接相互作用，从而起到保护数据的作用。那么在 PDO 中如何使用存储过程呢？

专家解答

1. 创建存储过程

这里首先创建一个存储过程，其 SQL 语句如下：

```
drop procedure if exists pro_reg;
delimiter //
create procedure pro_reg (in nc varchar(80), in pwd varchar(80), in email varchar(80),in address varchar(50))
begin
insert into tb_reg (name, pwd ,email ,address) values (nc, pwd, email, address);
end;
//
```

"drop"语句删除 MySQL 服务器中已经存在的存储过程 pro_reg。

"delimiter //"的作用是将语句结束符更改为"//"。

"in nc varchar(50)……in address varchar(50)"表示要向存储过程中传入的参数。

"begin……end"表示存储过程中的语句块，它的作用类似于 PHP 语言中的"{……}"。

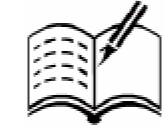
2. 在 PDO 中通过 CALL 语句调用创建存储过程

在 PDO 中通过 CALL 语句调用存储过程，实现用户注册信息的添加。其具体步骤如下：

创建 index.php 文件。首先，创建 form 表单，将用户注册信息通过 POST()方法提交到本页。然后，在本页中编写 PHP 脚本，通过 PDO 连接 MySQL 数据库，并且设置数据库编码格式为 UTF-8，获取表单中提交的用户注册信息。接着，通过 call 语句调用存储过程 pro_reg，将用户注册信息添加到数据表中。最后，通过 try{}catch{}语句块返回错误信息。其关键代码如下：

```
<?php
 if($_POST['submit']!=""){
 $dbms='mysql';                          //数据库类型，对于开发者来说，使用不同的数据
                                          库，只要修改它即可，不用记住那么多的函数
    $host='localhost';                   //数据库主机名
    $dbName='db_database';               //使用的数据库
    $user='root';                        //数据库连接用户名
    $pass='111';                         //对应的密码
    $dsn="$dbms:host=$host;dbname=$dbName";
    try {
        $pdo = new PDO($dsn, $user, $pass);      //初始化一个 PDO 对象，就是创建了数据库连
                                                  接对象$pdo
        $pdo->query("set names utf8");           //设置数据库编码格式
        $pdo->setAttribute(PDO::ATTR_ERRMODE,PDO::ERRMODE_EXCEPTION);
                                                 //定义错误异常模式
     $nc=$_POST['nc'];
        $pwd=md5($_POST['pwd']);
        $email=$_POST['email'];
        $address=$_POST['address'];
```





Note

```
                $query="call pro_reg('$nc','$pwd','$email','$address')";
                $result=$pdo->prepare($query);
                if($result->execute()){
                    echo "数据添加成功！ ";
                }else{
                    echo "数据添加失败！ ";
                }
            } catch (PDOException $e) {
                echo 'PDO Exception Caught.';
                echo 'Error with the database:<br/>';
                echo   'SQL Query: '.$query;
                echo '<pre>';
                echo "Error: " . $e->getMessage(). "<br/>";
                echo "Code: " . $e->getCode(). "<br/>";
                echo "File: " . $e->getFile(). "<br/>";
                echo "Line: " . $e->getLine(). "<br/>";
                echo "Trace: " . $e->getTraceAsString(). "<br/>";
                echo '</pre>';
            }
        }
    ?>
```

运行结果如图 9.8 所示。



图 9.8　通过存储过程完成用户的注册

专家点评

在 MySQL 中，创建存储过程的基本形式如下：

```
CREATE PROCEDURE sp_name ([proc_parameter[,…]])
[characteristic…] routine_body
```

参数说明：

❶ sp_name：表示存储过程的名称；

❷ proc_parameter：表示存储过程的参数列表；

❸ characteristic：表示指定存储过程的特性；

❹ routine_body：表示 SQL 代码的内容。

说明：

proc_parameter 中的参数由 3 部分组成，分别是输入输出类型、参数名称和参数类型。其形式为[IN | OUT | INOUT]param_name type。其中，IN 表示输入参数；OUT 表示输出参数；INOUT 表示既可以输入也可以输出；param_name 参数是存储过程的参数名称；type 参数指定存储过程的参数类型，该类型可以为 MySQL 数据库的任意数据类型。






MySQL 中存储过程的建立以关键字 create procedure 开始，后面仅跟存储过程的名称和参数。MySQL 的存储过程名称不区分大小写，如 PROCE1()和 proce1()代表同一存储过程名。存储过程的名称不能与 MySQL 数据库中的内建函数重名。

MySQL 存储过程的语句块以 begin 开始，以 end 结束。语句体中可以包含变量的声明、控制语句、SQL 查询语句等。由于存储过程内部语句要以分号结束，所以在定义存储过程前，应将语句结束标志“;”更改为其他字符，并且应降低该字符在存储过程中出现的机率，更改结束标志可以用关键字“delimiter”定义，例如：

```
mysql>delimiter //
```

存储过程创建之后，可用如下语句进行删除，参数 proc_name 指存储过程名。

```
drop procedure proc_name
```

问题 196　如何利用 PDO 连接 Oracle 11g?

问题阐述

在 PHP 中，如何利用 PDO 来连接 Oracle 11g 数据库呢？

专家解答

通过 PDO 连接 Oracle 11g 数据库，首先要在本机上安装 Oracle 11g 的客户端。然后，在 php.ini 文件中加载 extension=php_pdo.dll 和 extension=php_pdo_oci.dll 模块。最后，将 Oracle 11g 客户端的 oraociei11.dll、oci.dll 和 orannzsbb11.dll 复制到 Apache 的 bin 文件夹下。

注意：

切记在完成上述配置操作后，应重新启动 Apache 服务器，之后就可以通过 PDO 连接 Oracle 11g 数据库了。

应用示例

下面连接 Oracle 数据库服务器，数据库名称是 192.168.1.188:1521/orcl，数据库服务器的用户名是 system，密码是 mrsoft。

具体操作步骤为：首先定义数据库连接的参数，然后实例化 PDO 构造函数，最后通过 try catch 语句判断数据库是否连接成功。具体代码如下：

```
<?php
header("Content-type:text/html; charset=utf-8");
$dbms='oci';                          //数据库类型，对于开发者来说，使用不同的数据
                                        库，只要修改它即可，不用记住那么多的函数
$dbName='192.168.1.188:1521/orcl';    //使用的数据库
$user='system';                       //数据库连接用户名
$pass='mrsoft';                       //对应的密码
$dsn="$dbms:dbname=$dbName";
try {
    $pdo = new PDO($dsn, $user, $pass);   //初始化一个 PDO 对象，就是创建了数据库连接对
                                            象$pdo
    echo "Oracle 11g 连接成功!<br/>";
```





```
        $pdo = null;
    } catch (PDOException $e) {
        die ("Error!: " . $e->getMessage() . "<br/>");
    }
    ?>
```

连接测试结果如图 9.9 所示。

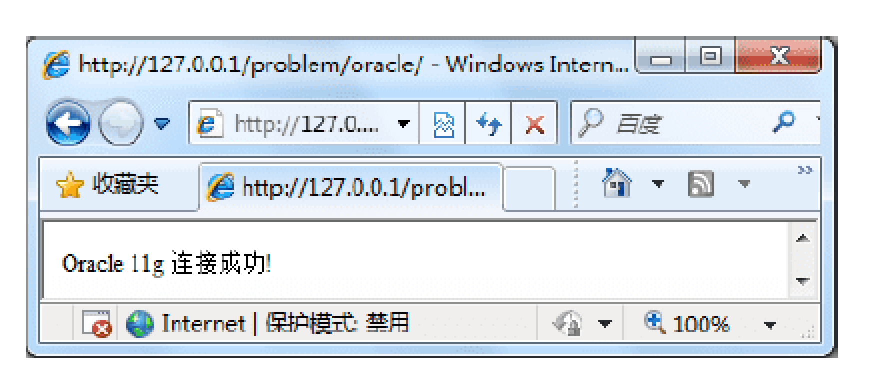


图 9.9　连接 Oracle 11g 数据库

专家点评

通过 PDO 连接 Oracle 数据库，连接 Oracle 10g 和连接 Oracle 11g 的方法是相同的，都必须在本机上安装 Oracle 数据库的客户端，然后将客户端中的 oraociei11.dll、oci.dll 和 orannzsbb11.dll 复制到 Apache 的 bin 文件夹下。

问题 197　如何封装 PDO 数据库连接类?

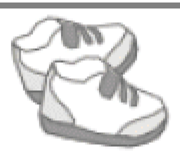
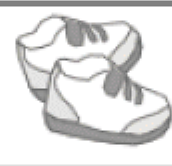
问题阐述

虽然 PDO 是数据库抽象层，可以连接、操作不同的数据库，但是就其连接方法而言，不同的数据库之间仍然存在一些区别，那么如何才能忽略这些区别，达到真正的统一呢？

专家解答

要想达到这种真正的统一，最有效的方法就是将数据库的连接方法封装到类中，直接向类中传递不同的参数，即可实现与不同数据库的连接。PDO 数据库连接类 ConnDB 的语法如下：

```
/************************************************
 * @ 说明：PDO 连接数据库类
 * @ 作者：明日科技
 * @ 学习社区：www.mrbccd.com
 * @ E-mail:mingrisoft@mingrisoft.com
************************************************/
class ConnDB{
    var $dbtype;
    var $host;
    var $user;
    var $pwd;
    var $dbname;
   /*
     * @ 方法说明：
     *    构造方法，类的初始化
     *
     * @ 参数说明：
     *    $dbtype：连接数据库的类型
     *    $host：数据库服务器主机名或 IP 地址
     *    $user：用户名
     *    $pwd：密码
     *    $dbname：数据库名称
     */
```


```
        function ConnDB($dbtype,$host,$user,$pwd,$dbname){
            $this->dbtype=$dbtype;
                $this->host=$host;
            $this->user=$user;
            $this->pwd=$pwd;
            $this->dbname=$dbname;
        }
        /*
         * @ 方法说明:
         *    获取数据库连接 ID
          */
        function GetConnId(){
            if($this->dbtype=="mysql" || $this->dbtype=="mssql"){    //判断数据库的类型，根据类
                                                                      型定义 DSN 的值
                    $dsn="$this->dbtype:host=$this->host;dbname=$this->dbname";
            }else{
                $dsn="$this->dbtype:dbname=$this->dbname";
            }
            try {
                    $conn = new PDO($dsn, $this->user, $this->pwd); //初始化 PDO 对象，创建数据
                                                                      库连接对象$pdo
                $conn->query("set names utf8");                       //设置编码格式
                    return $conn;                                     //返回数据库连接标识
            } catch (PDOException $e) {
                    die ("Error!: " . $e->getMessage() . "<br/>");
            }
        }
    }
```

通过 PDO 连接 MySQL 数据库，数据库名称为 db_database，用户名为 root，密码为 111。具体操作步骤为：首先封装 PDO 连接数据库类 system.class.inc.php，然后创建 system.inc.php 文件，载入 system.class.inc.php 文件，实例化数据库连接类 ConnDB，返回连接标识$conn。system.inc.php 文件的代码如下：

```
<?php
require("system.class.inc.php");                                    //包含数据库连接和操作类
$connobj=new ConnDB("mysql","localhost","root","111"," db_database");  //数据库连接类实例化
$conn=$connobj->GetConnId();                                        //执行连接操作，返回连接标识
?>
```

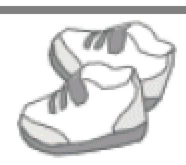
问题 198　如何封装 PDO 数据库操作类？

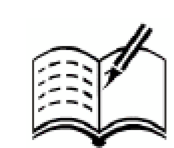
问题阐述

既然可以对 PDO 连接数据库的方法进行封装，那么也就可以对 PDO 操作数据库的方法进行封装，而这个 PDO 操作数据库的方法是如何实现的呢？

专家解答

PDO 数据库抽象层对数据库的操作方法进行了封装，实现对不同数据库的统一操作、管理。



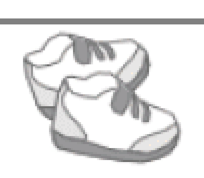


在我们封装的 PDO 操作数据库类中，它不但是不同数据库操作的集成，而且还可以对参数传递的 SQL 语句进行判断，进而根据不同的语句执行不同的操作，并且返回结果。PDO 数据库操作类 AdminDB 的语句如下：

```
/*************************************************
 * @ 说明：PDO 操作数据库类
 * @ 作者：明日科技
 * @ 学习社区：www.mrbccd.com
 * @ E-mail:mingrisoft@mingrisoft.com
*************************************************/
class AdminDB{
    /*
      * @ 方法说明：
      *    执行 SQL 语句
      *
      * @ 参数说明：
      *    $sql：所要执行的 SQL 语句
      *    $conn：数据库连接 ID
      */
    function ExecSQL($sqlstr,$conn){
        $sqltype=strtolower(substr(trim($sqlstr),0,6));
        $rs=$conn->prepare($sqlstr);                    //准备查询语句
        $rs->execute();                                 //执行查询语句，并返回结果集
        if($sqltype=="select"){
            $array=$rs->fetchAll(PDO::FETCH_ASSOC);     //获取结果集中的所有数据
            if(count($array)==0 || $rs==false)
                return false;                           //返回结果
            else
                return $array;                          //返回数组结果集
        }elseif ($sqltype=="update" || $sqltype=="insert" || $sqltype=="delete"){ //如果是此类语句
            if($rs)
                return true;                            //执行成功返回 true
            else
                return false;                           //否则返回 false
        }
    }
}
```

应用 PDO 操作数据库类，查询 db_database 数据库中 tb_commo 数据表中的数据，并且输出查询结果。具体操作步骤为：首先实例化 PDO 数据库操作类 AdminDB，然后通过实例化对象$admindb 调用 ExecSQL()方法，查询 tb_bigtype 数据表中的数据，最后输出查询结果。其关键代码如下：

```
$admindb=new AdminDB();                                          //数据库操作类实例化
$res = $admindb->ExecSQL("select * from tb_commo ",$conn,1,'1');    //执行查询语句
for($i=0;$i<count($res);$i++){
    <img src="<?php echo $res[$i]['pics'];?>" width="90" height="100" alt="<?php echo $res[$i]
['name'];?>" style="border: 1px solid #f0f0f0;" />
}
```



问题 199 如何封装 PDO 分页类？



问题阐述

连接、操作数据库的最终目的是数据的添加、更新、删除及查询。那么如何将 PDO 操作数据库查询出的结果集进行分页输出呢？是否可以将分页输出的方法封装到类中呢？

专家解答

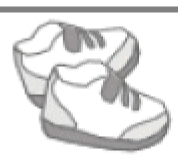
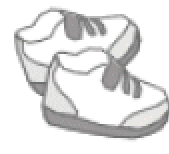
封装 SepPage 类，调用 PDO 操作数据库的方法，实现对数据库中数据的分页查询，并输出分页的超链接。SepPage 类的语法如下：

```
/***********************************************
 * @ 说明：PDO 分页类
 * @ 作者：明日科技
 * @ 学习社区：www.mrbccd.com
 * @ E-mail:mingrisoft@mingrisoft.com
***********************************************/
//分页类
class SepPage{
    var $rs;
    var $pagesize;
    var $nowpage;
    var $array;
    var $conn;
    var $sqlstr;
      /*
     * @ 方法说明：
     *    数据分页处理，并以数组形式返回分页信息
     *
     * @ 参数说明：
     *    $sqlstr：所要执行的 SQL 语句
     *    $conn：数据库连接 ID
     *    $pagesize：每页显示数据的条数
     *    $nowpage：当前显示的页数
        */
    function ShowData($sqlstr,$conn,$pagesize,$nowpage){        //定义方法
        if(!isset($nowpage) || $nowpage=="")                    //判断变量值是否为空
            $this->nowpage=1;                                   //定义每页起始页
        else
            $this->nowpage=$nowpage;
        $this->pagesize=$pagesize;                              //定义每页输出的记录数
        $this->conn=$conn;                                      //连接数据库返回的标识
        $this->sqlstr=$sqlstr;                                  //执行的查询语句
        $offset=($this->nowpage-1)*$this->pagesize;
        $sql=$this->sqlstr." limit $offset, $this->pagesize";
        $result=$this->conn->prepare($sql);                     //准备查询语句
        $result->execute();                                     //执行查询语句，并返回结果集
        $this->array=$result->fetchAll(PDO::FETCH_ASSOC); //获取结果集中的所有数据
```





```
            if(count($this->array)==0 || $this->array==false)
                return false;
            else
                return $this->array;
        }
        /*
         * @ 方法说明:
         *    创建数据分页输出的超链接
         *
         * @ 参数说明:
         *    $contentname: 超链接总称
         *    $utits: 超链接中数据的单位
         *    $anothersearchstr: 超链接传递的第一个参数
         *    $anothersearchstrs: 超链接传递的第二个参数
         *    $class: 超链接传递的样式
            */
        function ShowPage($contentname,$utits,$anothersearchstr,$anothersearchstrs,$class){
            $str="";
            $res=$this->conn->prepare($this->sqlstr);              //准备查询语句
            $res->execute();                                       //执行查询语句，并返回结果集
            $this->array=$res->fetchAll(PDO::FETCH_ASSOC);  //获取结果集中的所有数据
            $record=count($this->array);                           //统计记录总数
            $pagecount=ceil($record/$this->pagesize);              //计算共有几页
            $str.=$contentname." ".$record." ".$utits."  每 页  ".$this->pagesize."
 ".$utits." 第 ".$this->nowpage." 页/共 ".$pagecount." 页";
            $str.="    ";
            if($this->nowpage!=1)
                $str.="<a
href=".$_SERVER['PHP_SELF']."?page=1&page_type=".$anothersearchstr."&parameter2=".$anothersearchstrs.
" class=".$class.">首页</a>";
            else
                $str.="<font color='#555555'>首页</font>";
            $str.=" ";
            if($this->nowpage!=1)
                $str.="<a
href=".$_SERVER['PHP_SELF']."?page=".($this->nowpage-1)."&page_type=".$anothersearchstr."&parameter2=".
$anothersearchstrs." class=".$class.">上一页</a>";
            else
                $str.="<font color='#555555'>上一页</font>";
            $str.=" ";
            if($this->nowpage!=$pagecount)
                $str.="<a
href=".$_SERVER['PHP_SELF']."?page=".($this->nowpage+1)."&page_type=".$anothersearchstr."&parameter2=".
$anothersearchstrs." class=".$class.">下一页</a>";
            else
                $str.="<font color='#555555'>下一页</font>";
            $str.=" ";
            if($this->nowpage!=$pagecount)
                $str.="<a
href=".$_SERVER['PHP_SELF']."?page=".$pagecount."&page_type=".$anothersearchstr."&parameter2=".$anot
hersearchstrs." class=".$class.">尾页</a>";
            else
```

```
                $str.="<font color='#555555'>尾页</font>";
            if(count($this->array)==0 || $this->array==false)
                return "无数据！ ";
            else
                return $str;
        }
    }
```

应用 PDO 分页类，对 db_database 数据库中 tb_commo 数据表中的数据进行分页输出。载入类的实例化文件，通过分页类 AdminDB 的$admindb 对象调用 ShowData ()方法，完成数据表中数据的分页查询操作，调用 ShowPage()方法输出分页超链接。其关键代码如下：

```
    $connobj=new ConnDB("mysql","localhost","root","111","db_database");    //数据库连接类实例化
    $conn=$connobj->GetConnId();                                  //执行连接操作，返回连接标识
    $admindb=new AdminDB();                                       //数据库操作类实例化
    $seppage=new SepPage();                                       //分页类实例化
    if(isset($_GET['page'])){                                     //判断分页变量是否存在
        $page=$_GET['page'];
    }else{
        $page=1;
    }
    $res = $seppage->ShowData("select * from tb_commo ",$conn,2,$page);    //执行查询语句
    for($i=0;$i<count($res);$i++){                                         //循环输出查询结果
        <table width="636" height="134" border="0" cellspacing="0" cellpadding="0">
                <tr>
                <td width="145" rowspan="5" align="center" valign="middle"><img src="<?php echo
$res[$i]['pics'];?>" width="90" height="100" alt="<?php echo $res[$i]['name'];?>" style="border: 1px solid
#f0f0f0;" /></td>
                <td height="35">商品名称：<?php echo $res[$i]['name'];?></td>
                <td width="156" height="35">商品类别：<?php echo $res[$i]['class'];?></td>
                <td width="157">商品型号：<?php echo $res[$i]['model'];?></td>
              </tr>
            <tr>
                <td height="25"><?php echo $seppage->ShowPage("产品","个",$_GET ['page_type'],
","a");?></td>
              </tr>
        </table>
    <?php
    }
    ?>
```

问题 200　如何封装 ADODB 连接数据库类?

问题阐述

连接数据库类的封装意味着通过 ADODB 连接不同的数据库的方法更加简单，更容易更新和修改。只要向类中传递对应的参数值，即可完成与不同数据库服务器的连接。那么如何实现 ADODB 连接数据库类的封装呢？





专家解答

ADODB 连接数据库系统使用 ADONewConnection()函数，连接数据库如果选择持久化连接，则使用 PConnect()函数；如果选择非持久化连接，则使用 Connect()函数。

（1）使用 ADONewConnection()函数连接数据库系统。其语法如下：

```
ADONewConnection($databaseType)
```

参数$databaseType：表示要连接的数据库系统的名称，例如“mysql”、“mssql”等。

（2）使用 PConnect()函数实现与数据库持久化连接。其语法如下：

```
PConnect($host,[$user],[$password],[$database])
```

参数说明：

❶ $host：数据库系统的服务器所在地址。如果是本机操作，则参数值为 localhost。

❷ $user：数据库用户名。

❸ $password：数据库密码。

❹ $database：使用到的数据库。

（3）使用 Connect()函数实现与数据库的非持久化连接。其语法如下：

```
Connect($host,[$user],[$password],[$database])
```

参数说明：

❶ $host：数据库系统的服务器所在地址。如果是本机操作，则参数值为 localhost。

❷ $user：数据库用户名。

❸ $password：数据库密码。

❹ $database：使用到的数据库。

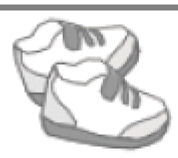
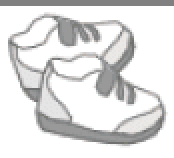
掌握连接数据库服务器所使用的函数之后，就是定义类、方法和参数，完成与指定数据库服务器的连接，并返回连接标识。

应用示例

通过 AODOB 封装数据库连接类，实现与 MySQL 数据库的连接。其具体实现步骤如下：

（1）首先创建 found.database.php 文件，定义 ConnDB 连接数据库类，定义 ConnDB()方法为成员变量赋值；然后定义 GetConnId()方法实现与不同数据库服务器的连接，并返回连接标识；最后定义 CloseConnId()方法关闭数据库连接。其关键代码如下：

```
<?php
//数据库连接类
class ConnDB{
        var $dbtype;                    //数据库服务器类型
        var $host;                      //服务器名称
        var $user;                      //用户名
        var $pwd;                       //密码
        var $dbname;                    //数据库名称
        var $debug;                     //是否返回执行的 SQL 语句，默认设置为返回 SQL
        var $conn;                      //连接数据库时返回的连接标识
        function ConnDB($dbtype,$host,$user,$pwd,$dbname,$debug=true){
                                        //构造方法，为成员变量赋值
            $this->dbtype=$dbtype;
            $this->host=$host;
```

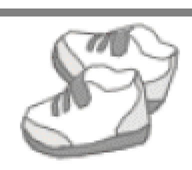
```
            $this->user=$user;
            $this->pwd=$pwd;
            $this->dbname=$dbname;
            $this->debug=$debug;
        }
        function GetConnId(){                  //实现与不同数据库的连接并返回连接对象
          require("../adodb5/adodb.inc.php");             //调用 ADODB 类库文件
                if($this->dbtype=="mysql" || $this->dbtype=="mssql"){
                                                          //判断成员变量传递的数据库类型
                    if($this->dbtype=="mysql")            //判断如果是 MySQL 数据库
                            $this->conn=NewADOConnection("mysql");
                                                          //执行与 MySQL 数据库的连接
                    else
                            $this->conn=NewADOConnection("mssql");
                        $this->conn->Connect($this->host,$this->user,$this->pwd,$this->
dbname);                                                       //数据库连接的用户、密码
                }elseif($this->dbtype=="access"){              //判断如果使用的是 Access 数据库
                        $this->conn=NewADOConnection("access");
                        $this->conn->Connect("Driver={Microsoft Access Driver (*.mdb)};
Dbq=".$this->dbname.";Uid=".$this->user.";Pwd=".$this->pwd.";");   //执行连接 Access 数据库
                }
                $this->conn->Execute("set names utf8");        //设置数据库的编码格式
                if($this->dbtype=="mysql")
                        $this->conn->debug=$this->debug;
                    return $this->conn;                        //返回连接对象
        }
        function CloseConnId(){                                //定义关闭数据库的方法
            $this->conn->Disconnect();                         //执行关闭的操作
        }
    }
    ?>
```

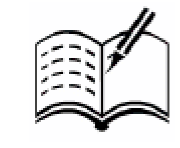
（2）创建 connect.database.php 文件，实例化数据库连接类 ConnDB，并向类中传递对应的参数，完成与 MySQL 数据库服务器中 db_database 数据库的连接。其关键代码如下：

```
<?php
require("found.database.php");                                 //包含类文件
$connobj=new ConnDB("mysql","localhost","root","111","db_database",false); //数据库连接类实例化
$conn=$connobj->GetConnId();                                   //返回连接标识
?>
```

（3）创建 index.php 文件，包含数据库连接类的实例化文件，根据返回的连接标识判断数据库是否连接成功。其关键代码如下：

```
<?php
include_once 'conn/connect.database.php';                      //包含数据库连接类的实例化文件
if($conn){
    echo "<script>alert('数据库连接成功！');window.location.href='http://www.mrbccd.com';</script>";
}else{
    echo "<script>alert('数据库连接失败！ ');window.location.href='index.php';</script>";
}
?>
```





运行本示例，连接 MySQL 数据库服务器，选择 db_database 数据库，其运行结果如图 9.10 所示。





图 9.10　ADODB 连接数据库类

专家点评

定义连接数据库类，在定义连接方法 GetConnId()时，必须先通过 require()语句包含 ADODB 类库文件 adodb.inc.php，否则其他一切操作都无从谈起。

建议在程序的开发阶段，在运用连接数据库操作类时，将参数$debug 的值设置为 True，这样可以返回执行的 SQL 语句，便于获取 SQL 语句在执行过程中出现的错误。如果是在生产环境中运用连接数据库类，那么就要将$debug 的值设置为 False。

问题 201　ADODB 操作结果集的方法有哪些?

问题阐述

通过 ADODB 类库操作数据库，不但要掌握数据库的连接方法，并且还要掌握操作结果集的方法。那么 ADODB 操作结果集有哪些方法呢?

专家解答

当使用 execute()函数执行 SQL 指令时，会回传一个 ADORecordSet 对象。通过对 ADORecordSet 对象的控制，可以对结果集进行很多操作。其操作结果集的方法如下：

1. fields

该变量保存的是当前指针所指向的记录。

2. EOF

该变量记录当前指针是否指向最后一条记录。如果是最后一条记录，则设定为 True；否则为 False。

3. GetArray([$number_of_rows])

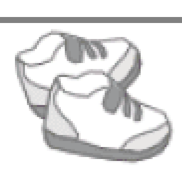
返回从当前指针指向的记录开始，到（$number_of_rows - 1）行的全部记录的数组。参数$number_of_rows 是指定的记录行。如果没有给出，则一直到 EOF 才停止。

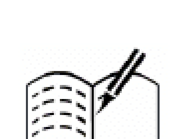
4. MoveNext()

函数作用：将 ADORecordSet（结果集）的指针下移一位。如果成功，则返回 True，否则返回 False。

5. Move($to)

函数作用：将 ADORecordSet（结果集）的指针移动到指定位置。如果$to 等于 0，则指针指向结果集的第一条数据；如果$to 的值大于结果集，则指针指向最后一条数据。注意：这里





变量$to 只能是绝对定位。

6. MoveFirst()

函数作用：指针移动到第一条数据。等同于 Move(0)。

7. MoveLast()

函数作用：指针移动到最后一条数据。等同于 Move(RecordCount() - 1)。

8. FetchRow()

函数作用：返回当前指针指向的记录的数组，如果是 EOF，回传 False。FetchRow 不要和 MoveNext()混用。

9. FetchField($column_number)

函数作用：返回一个对象，包含着字段$column_number 的名称、空间长度等相关信息。
参数$column_number 是要查看的字段名称。

10. FetchNextObject([$toupper=true])

函数作用：返回当前指针所指向的记录的对象形式，并且指针自动下移一行。
参数$toupper 如果等于 True，则字段名为大写形式。

11. FieldCount()

函数作用：返回结果集中的字段数。

12. RecordCount( )

函数作用：返回结果集中的记录数。

13. MetaType($nativeDBType [,$field_max_length],[$fieldobj])

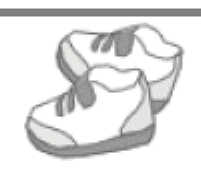
函数作用：标准化不同数据库系统下的数据类型的表示法。处理不同的数据库系统有一个问题：每一个数据库系统对于相同的数据类型会有不同的表示法，如支持超长文本的类型，有的表示为 text，有的表示为 long character，还有的表示为 clob 等。MetaType()函数可以将这几种表示法统一起来。如用大写字母 D 来表示所有的日期类型，用大写字母 T 表示所有的时间类型。

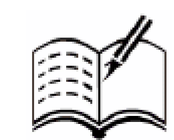
参数说明：

❶ $nativeDBType：数据表中的数据类型。
❷ $fields_max_length：字段最大长度。
❸ $fieldsobj：字段对象。

MetaType()支持的标准化数据类型有：

☑ C：支持 char、varchar 等字符类型
☑ X：支持 text、long character 等长类型。
☑ B：blob 或 binary image。
☑ D：日期 date。
☑ T：时间 timestamp。
☑ L：逻辑类型和位类型。
☑ N：包含编号、整型、浮点型等数字类型。
☑ R：包含序列、自动增加整数等序列类型。





应用示例

在本示例中应用 ADODB 类库中操作结果集的方法，循环输出数据表中的字段和值。其具体步骤如下：

（1）创建 conn 文件夹，编写 conn.php 文件，载入 adodb.ini.php 文件。实现与 MySQL 数据库服务器中 db_database 数据库的连接，并设置数据库的编码为 UTF-8。

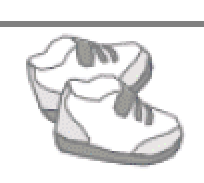
（2）创建 index.php 文件，包含数据库连接文件，定义 SQL 查询语句，通过 execute()函数执行查询操作，通过 FieldCount()函数统计字段数量，应用 for 语句和 FetchField()函数循环输出各个字段的值，并应用 metaType()函数对字段中的数据类型进行判断，同时应用 DBDate()函数对时间进行格式化。其关键代码如下：

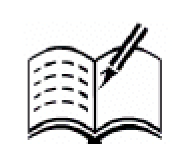
```
<?php
include_once 'conn/conn.php';                                       //载入数据库连接文件
$count = 1;
$sqlstr = 'select * from tb_object where id=' . $count;             //sql 查询语句
$rst = $conn->execute ( $sqlstr ) or die ( 'error：  ' . $conn->errorMsg () );   //执行查询语句
if (false != $rst) {                                                //如果有查询结果
        for($i = 0; $i < $rst->FieldCount (); $i ++) {              //循环输出各个字段
            $fields = $rst->FetchField ( $i );                      //生成字段信息对象
            $type = $fields->type;
    ?>
    <td bgcolor="#FFFFFF">
<?php
//从对象中获取字段的类型信息
if ($rst->metaType ( $type, - 1, $fields ) == "T")                  //如果标准类型为“T”
        echo $conn->DBDate ( $rst->fields [$i] );                   //使用 DBDate 函数格式化时间
    else
        echo $rst->fields [$i];                                     //如果是其他类型，则直接输出
    ?>
</td>
<?php
    }
    }
    ?>
```

运行结果如图 9.11 所示。



图 9.11　ADODB 操作结果集的方法





专家点评

在本实例中只是通过 FieldCount()函数和 FetchField()函数对数据表中的字段进行操作。在后面的问题中将应用 EOF、fields 和 movenext()函数以及 while 语句完成数据表中数据的循环输出，其实现的是结果集中行的操作。

问题 202　ADODB 中如何控制结果集的存取方法?

问题阐述

在 ADODB 中如何控制结果集的存取方法?

专家解答

在通过对 SQL 语句的操作之后，多数都会返回一个结果集（ADORecordSet 对象）。通过对 ADORecordSet 对象的控制，可以对结果集进行各项操作，这里介绍的是如何控制结果集的存取方法。

1. $ADODB_COUNTRECS

当该变量为 true 时，RecordCount()函数会在数据库驱动不支持 select 指令返回的记录总数时进行自动模拟，返回记录数，默认值为 true。每次进行查询操作时都会自动检查该变量的值。由于这个功能十分消耗内存，所以不建议使用。

2. $ADODB_CACHE_DIR

该变量设置缓存目录，一般在使用缓存函数时会用到该变量，如 CacheExecute()函数。

3. $ADODB_FETCH_MODE

该变量决定结果集以哪种方式进行存取，其包括 4 个值：

（1）define('ADODB_FETCH_DEFAULT',0);

（2）define('ADODB_FETCH_NUM',1)

结果集按照字段序号为索引进行存取，如$rst -> fields[0]、$rst -> fields[1]。

（3）define('ADODB_FETCH_ASSOC',2)

结果集按照字段名称为索引进行存取，如$rst -> fields[bigclass]、$rst -> fields[id]。

（4）define('ADODB_FETCH_BOTH',3)

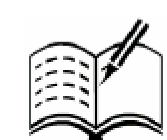
结果集同时支持第（2）、第（3）两种方式。如果未设置该变量，则默认值为（1），不同类型的数据库驱动，默认值也不相同。这里不建议使用变量（4），因为有很多的驱动并不支持。

4. 应用示例

在本示例中应用 EOF、fields 和 movenext()函数以及 while 语句完成数据表中数据的循环输出，并且应用$ADODB_FETCH_MODE 控制结果集的存储方式。其具体步骤如下：

（1）创建 conn 文件夹，编写 conn.php 文件，载入 adodb.ini.php 文件。实现与 MySQL 数据库服务器中 db_database 数据库的连接，应用$ADODB_FETCH_MODE 设置结果集以字段名称为索引进行存取，并设置数据库的编码为 UTF-8。其代码如下：





```
<?php
include_once ('../adodb5/adodb.inc.php');                    //载入 adodb.ini.php 文件
$conn = ADONewConnection('mysql');                           //连接 MySQL 数据库
$conn -> PConnect('localhost','root','111','db_database');   //连接 db_database 库
$ADODB_FETCH_MODE = ADODB_FETCH_ASSOC;                       //设置结果集以字段名称为索引
$conn -> execute('set names utf8');                          //设置编码

?>
```

（2）创建 index.php 文件，完成数据的循环输出。首先，包含数据库连接文件。然后，定义 SQL 查询语句，通过 execute()函数执行查询操作。最后，while 循环输出查询结果。其关键代码如下：

```
<?php
    include_once 'conn/conn.php';                               //载入数据库连接文件
    $sqlstr = 'select * from tb_user where id limit 5';         //sql 查询语句
    $rst = $conn->execute ( $sqlstr ) or die ( 'error：  ' . $conn->errorMsg () );    //执行查询语句
    while ( ! $rst->EOF ) {                                      //wihle 语句循环输出结果
?>
<tr>
        <td width="126" height="30" align="center"><?php echo $rst->fields ['user']; ?></td>
        <td width="117" align="center"><?php echo $rst->fields ['dates']; ?></td>
</tr>
<?php
        $rst->movenext ();                                       //指针下移
    }
    $rst->close ();                                              //关闭连接
    $conn->close ();
    ?>
```

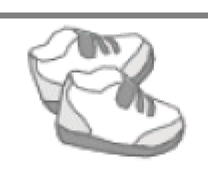
运行结果如图 9.12 所示。



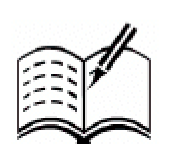
图 9.12　ADODB 类库中对数据库中数据输出的控制

专家点评

在专家解答中已经对 ADODB 类库中结果集的存取方式进行了详细介绍，至于如何选择，需要根据实际情况具体问题具体分析，也可以不设置结果集的存取方式。建议读者使用字段名称为索引进行数据存取，这样在获取数据值时更准确一些，不会因为数据表中字段的添加或删除而影响到获取的值。



问题 203　如何一次连接两个数据库？



Note

问题阐述

在实际的程序开发中，如果要实现数据在不同数据库服务器之间的转换，最理想的方法就是编写一个数据转换的程序，那么如何编写该程序呢？

专家解答

一次连接两个数据库，即同时执行两次 ADONewConnection()函数和 PConnect()函数，分别指定不同 MySQL 数据库服务器和 Access 数据库服务器。

在连接 MySQL 服务器时，设置的用户名是 root，密码是 111，指定的数据库是 db_database，设置的编码格式是 UTF-8。

在连接 Access 服务器时，指定的数据库是 db_database.mdb，注意这里设置的必须是 Access 数据库文件在服务器中的绝对路径。

同时连接两个数据库的代码如下：

```
<?php
include_once('../adodb5/adodb.inc.php');                    //载入（include）adodb.inc.php 文件
$conn = ADONewConnection('mysql');                          //建立连接
$conn -> PConnect('localhost','root','111','db_database');  //连接数据库
$conn -> execute('set names utf8');                         //设置编码格式
$con = ADONewConnection("access");                          //连接 Access 数据库
$con-> PConnect("Driver={Microsoft Access Driver (*.mdb)};Dbq=F:\\AppServ\\www\\mr\\13\\008\\
conn\\data\\db_database.mdb");
$con-> execute('set names gb2312');                         //设置编码格式
?>
```

在将从 MySQL 数据库中读取的数据添加到 Access 数据库时，应用的是 ADODB 类库中的 GetInsertSQL()函数。使用该函数可向数据库中添加新的记录，其语法如下：

```
GetInsertSQL(&$rs, $arrFields)
```

参数$rs 设置要添加记录的结果集；参数$arrFields 设置新记录的字段值及字段名称。

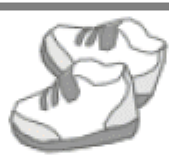
应用示例

本示例同时连接两个不同的数据库服务器，完成数据由 MySQL 数据库到 Access 数据库的转换。其具体步骤如下：

（1）创建 conn 文件，编写 conn.php 文件，完成与两个数据库服务器的连接，其方法可参考专家解答。

（2）创建 index.php 文件，将从 MySQL 数据库中读取出的数据通过 GetInsertSQL()函数添加到 Access 数据库中，并且在本页中输出 MySQL 数据库中的数据，同时创建数据导出和查看 Aceess 数据库中数据的超链接。其关键代码如下：

```
<?php
include ("conn/conn.php");
$sql = "select * from tb_user where id limit 5 ";           //创建 sql 语句
```



```
        $rst = $conn->execute ( $sql ) or die ( 'error:  ' . $conn->errorMsg () );      //执行查询语句
        $array = array ();
        while ( ! $rst->EOF ) {
                $array ['id'] = $rst->fields ['id']; # process $arr
                $array ['user'] = iconv ( "utf-8", "gb2312", $rst->fields ['user'] ); # process $arr
                $array ['pass'] = $rst->fields ['pass']; # process $arr
                $array ['dates'] = $rst->fields ['dates']; # process $arr
                if ($_GET ['insert'] == "导出") {
                    $sqls = "select * from tb_user where id=0 ";                        //创建 sql 语句
                    $ret = $con->execute ( $sqls ) or die ( 'error:  ' . $con->errorMsg () ); //执行查询语句
                    $insert = $con->getInsertSQL ( $ret, $array ) or die ( 'update error:  ' . $con->errorMsg
() ); //添加新数据
                    $con->execute ( $insert );
                }
        ?>
        <tr>
                <td><?php echo $rst->fields ['id']; ?></td>
                <td height="20"><?php echo $rst->fields ['user']; ?></td>
                <td><?php echo $rst->fields ['dates']; ?></td>
            </tr>
        <?php
                $rst->movenext ();                                                      //指针下移
            }
            $rst->close ();                                                             //关闭连接
            $conn->close ();
            ?>
```

（3）创建 access.php 文件，包含数据库连接文件 conn.php，完成 Access 数据库中数据的循环输出。

运行本示例，单击图 9.13 中的“导出”按钮，将 MySQL 数据库中的数据导入 Access 数据库中，单击“查看 ACCESS”超链接将看到如图 9.14 所示的存储于 Access 数据库中的数据。



图 9.13　MySQL 数据库中的数据



图 9.14　Access 数据库中的数据

专家点评

通过 ADODB 连接 Access 数据库时，在设置数据库文件的存储位置时，必须指定数据库



文件在服务器中的绝对路径，同时为了确保 Access 数据库文件的安全，可以为 Access 数据库设置密码。通过选择 Access 菜单栏中的“工具”→“安全”→“设置数据库密码”命令即可设置 Access 数据库的密码。



Note

若想撤销数据库密码，只需选择 Access 菜单栏中的“工具”→“安全”→“撤销数据库密码”命令，在弹出的“撤销数据库密码”对话框中输入设置时的密码即可。

问题 204　如何在 ADODB 中生成 HTML 表格？

问题阐述

ADODB 如何生成 HTML 表格？

专家解答

在 ADODB 类库中，有一种方法可以返回表格式的结果集，即将查询结果直接以表格的形式进行输出。这种方法就是 rs2html()函数，其返回一个 HTML 表格格式的结果集。其语法如下：

```
function rs2html($adorecordset,[$tableheader_attributes], [$col_titles])
```

参数说明：

❶ $adorecordset：要返回的结果集。

❷ $tableheader_attributes：对表格的参数及属性设置。

❸ $col_titles：对字段的重新命名。

要运用本函数，必须载入 adodb 类库中的 tohtml.inc.php 文件。

应用示例

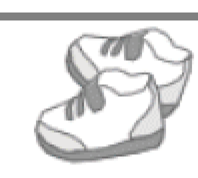
（1）创建 conn 文件夹，编写 conn.php 文件，完成与 MySQL 数据库的连接。

（2）创建 index.php 文件，包含数据库连接文件，定义 SQL 查询语句，应用 rs2html()函数将查询结果以表格的形式输出。其关键代码如下：

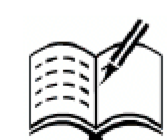
```
<?php
    include_once 'conn/conn.php';                          //载入数据库连接文件
    include_once '../adodb5/tohtml.inc.php';               //载入 tohtml.inc.php 文件
    $rst = $conn -> execute('select * from tb_bccd');       //返回查询结果集
    rs2html($rst,' width="775" border="1" cellpadding="1" cellspacing="1" bordercolor="#FFFFFF"
bgcolor="#CCCCCC"',array('序号','名称','类别','时间'));
?>
```

专家点评

rs2html()函数是一个独立的函数(rs2heml = recordset to html)，相当于 PHP 中的 odbc_result_all 函数。它输出一个完整的 adorecordSet，$adorecordset 如同一个 HTML 表格。$tableheader_attributes 允许控制表格中的参数，如 cellpadding、cellspacing 或 border 等属性。最后，通过 $col_titles 参数设置数据表中字段的名称。注意，它只是一个快速除错的机制，并不是一个理想的结果集输出方法。



问题 205　如何在 ADODB 中生成下拉列表框?



问题阐述

ADODB 如何生成下拉列表框?

专家解答

ADODB 中生成下拉列表框应用的是 GetMenu()函数。

GetMenu()函数返回一个 HTML 下拉列表框（select）。结果集的第一列（fields[0]）被显示到下拉列表中；第二列（fields[1]）被设置为<option>标签的 value 属性。其语法如下：

```
GetMenu($name, [$default_str=''], [$blank1stItem=true], [$multiple_select=false], [$size=0], [$moreAttr=''])
```

参数说明：

❶ $name：下拉列表框的名称。

❷ $default_str：如果$default_str 的值等于其中一个 fields[0]，那么这个 fields[0]就是该下拉列表的默认选项，相当于为这个 fileds[0]添加了属性<selected>。

❸ $blank：该参数默认值为 true。在生成下拉列表框时，第一行选项默认为空，即<option></option>。如果为 false，则没有默认的空选项。

❹ $multiple_select：是否支持多选。默认值为 false，即不允许。

❺ $size：列表框选取的大小。默认为 0，生成一个下拉列表。如果$size 的值大于 0，则生成一个列表框。

❻ $moveAttr：该参数可以增加列表框的属性。例如添加一个 JavaScript 脚本等。

应用示例

应用 ADODB 中的 GetMenu2()函数将查询结果集中的数据直接输出到下拉列表框中。其具体步骤如下：

（1）创建 conn 文件夹，编写 conn.php 文件，完成与 MySQL 数据库的连接。

（2）编写 index.php 文件，创建 form 表单，设置 action 属性值为 index_ok.php。包含数据库连接文件，定义 SQL 查询语句，应用 GetMenu2()函数将查询结果以下拉列表框的形式输出。其关键代码如下：

```
<?php
include_once 'conn/conn.php';                          //载入数据库连接文件
$sqlstr = 'select name,id from tb_bccd';               //sql 查询语句
$rst = $conn -> execute($sqlstr) or die('execute error:   '.$conn -> errorMsg());
                                                       //执行查询语句，返回结果集
?>
<!--   调用 getMenu2()函数   -->
    <?php echo $rst -> getMenu2('type',$_POST['name'],false); ?>
```

运行结果如图 9.15 所示。

（3）创建 index_ok.php 文件，获取表单中提交的数据，完成用户的登录操作。

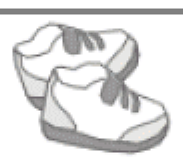
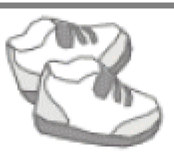





图 9.15　ADODB 生成下拉列表框

专家点评

不知道读者是否注意到，在本示例中运用的并不是专家解答中讲解的 GetMenu()函数，而是 GetMenu2()函数。GetMenu2()函数的作用与 GetMenu()函数基本相同，其区别在于：GetMenu()中的参数$default_str 和 fields[0]比较，如果相等，那么 fields[0]就是这个下拉列表的默认选项；而 getMenu2()中的参数$default_str 则是和 fields[1]进行比较，其他不变。

问题 206　一个小巧的分页函数——ADODB_Pager()

问题阐述

在 ADODB 中，有一种简单的数据分页输出的方法，它被封装到一个函数中，这就是 ADODB_Pager()函数。那么这个函数是如何使用的呢？

专家解答

ADODB_Pager()函数是 ADODB_Pager 类的构造函数，通过类中的 render()函数可以实现分页功能。

ADODB_Pager()函数的语法如下：

```
ADODB_Pager($conn, $sql, $id = 'adodb', $showPageLinks = false)
```

参数说明：

❶ $conn：是数据库连接对象；

❷ $sql：是执行的 SQL 语句；

❸ $id：每个分页的 id 号；

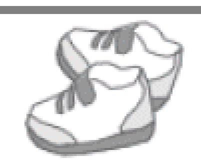
❹ $showPageLinks：设置是否显示各个页的链接，默认为 false。

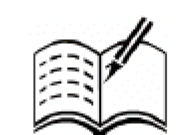
应用示例

应用 ADODB_Pager()函数实现一个分页功能。其具体步骤如下：

（1）创建 index.php 文件，包含数据库连接文件 conn.php，包含 adodb5 中的 adodb-pager.inc.php 文件，实例化 ADODB_Pager 类，传递查询参数，设置每页显示 10 条记录。代码如下：

```
<?php
include 'conn/conn.php';                              //载入数据库连接文件
include_once '../adodb5/adodb-pager.inc.php';          //载入 adodb_parper.ini.php 文件
/*   声明对象   */
$pager = new ADODB_Pager($conn,"select id,user as '用户名', pass as '密码',dates as '注册时间'
```





```
from tb_user");
            /*   调用$Render 函数确定每页显示多少    */
            $pager -> Render(10);
            ?>
```

（2）打开 adodb5 文件夹下的 adodb-pager.inc.php 文件，重新设置页面的样式。运行结果如图 9.16 所示。

id	用户名	密码	注册时间
67	VC	789456	Sat 03, Apr 2010
68	ASP.NET	456123	Sat 03, Apr 2010
69	JAVA WEB	789456	Sat 03, Apr 2010
70	C#	456123	Sat 03, Apr 2010
71	mr	fdb390e945559e74475ed8c8bbb48ca5	Fri 02, Apr 2010
72	明日科技	123456	Fri 02, Apr 2010
73	php	123456	Sat 03, Apr 2010
74	java	456789	Sat 03, Apr 2010
75	JavaScript	789456	Sat 03, Apr 2010
76	VB	456123	Sat 03, Apr 2010

当前页码 7/12　首页　上一页　下一页　尾页

图 9.16　应用 ADODB_Pager()函数实现一个分页功能

专家点评

应用 ADODB_Pager()函数实现分页功能有一个前提条件，就是在包含 ADODB 中的 adodb.inc.php 文件的同时还要包含 adodb-pager.inc.php 文件。

问题 207　如何通过事务处理机制保证数据库操作的完整性？

问题阐述

事务处理机制在程序开发过程中有着非常重要的作用，它可以使整个系统更加安全。如在银行处理转账业务时，如果 A 账户中的金额刚被转出，而 B 账户还没来得及接收就发生停电事故，这会给银行和个人带来很大的经济损失。采用事务处理机制，一旦在转账过程中发生意外，则程序将回滚，不做任何处理，那么在 ADODB 中是如何实现事务处理的呢？

专家解答

在 ADODB 中，事务处理应用的是如下 3 个函数。

（1）BeginTrans()函数：开启事务处理。该函数和接下来的两个函数都是应用在 PHP 的事务处理中。

（2）CommitTrans()函数：如果成功完成数据库操作，则执行该函数。

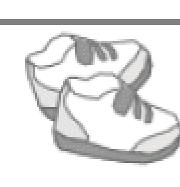
（3）RollbackTrans()函数：结束一次操作，恢复操作前的所有改变。如果成功则返回 True；如果服务器不支持则返回 False。

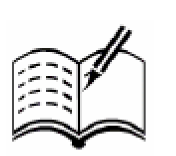
应用示例

本示例实现数据在不同数据库间的转换功能，在将 MySQL 数据库中数据转换到 Access 数据库后，删除 MySQL 数据库中的原始数据，并应用事务处理机制保证数据转换操作安全、顺利地进行。其具体实现步骤如下：

（1）创建 conn 文件夹，编写 conn.php 文件，完成与 MySQL 数据库的连接。

（2）首先，编写 index.php 文件，包含数据库连接文件，定义 SQL 语句，循环输出 MySQL 数据库中的数据，通过 BeginTrans()函数开启事务处理。然后，根据超链接传递的参数进行判断，通过 getInsertSQL()函数执行数据的转移操作，转移成功后通过 delete 语句删除原始数据库中的数据。最后，根据添加语句和删除语句的执行情况，判断是执行 CommitTrans ()提交操作





还是执行 RollbackTrans ()回滚操作。其关键代码如下：

```
<?php
include ("conn/conn.php");
$sql = "select * from tb_user where id order by id limit 5 ";                    //创建 sql 语句
$rst = $conn->execute ( $sql ) or die ( 'error:   ' . $conn->errorMsg () );        //执行查询语句
$array = array ();
$con->BeginTrans ();                                                             //开始事务处理
$conn->BeginTrans ();                                                            //开始事务处理
while ( ! $rst->EOF ) {
        if ($_GET ['insert'] == "数据转移") {
            $array ['id'] = $rst->fields ['id']; # process $arr
            $array ['user'] = iconv ( "utf-8", "gb2312", $rst->fields ['user'] ); # process $arr
            $array ['pass'] = $rst->fields ['pass']; # process $arr
            $array ['dates'] = $rst->fields ['dates']; # process $arr
            $sqls = "select * from tb_user where id=0 ";                         //创建 sql 语句
            $ret = $con->execute ( $sqls ) or die ( 'error: ' . $con->errorMsg () );    //执行查询语句
            $insert = $con->getInsertSQL ( $ret, $array ) or die ( 'update error:' . $con->errorMsg () );
                                                                                 //添加新数据
            $result=$con->execute ( $insert );
            $sql = 'delete from tb_user where id =' . $rst->fields ['id'];       //sql 删除语句
            $results = $conn->execute ( $sql ) or die ( 'execute error:   ' . $conn->ErrorMsg () );
                                                                                 //执行删除语句
        }
    ?>
        <tr>
                <td><?php echo $rst->fields ['id']; ?></td>
                <td height="20"><?php echo $rst->fields ['user']; ?></td>
                <td><?php echo $rst->fields ['dates']; ?></td>
            </tr>
<?php
        $rst->movenext ();                                                       //指针下移
}
if($result){
        $con->CommitTrans ();                                                    //执行提交
        if ($results) {
            $conn->CommitTrans ();                                               //执行提交
        } else {
            $conn->RollbackTrans ();                                             //执行回滚操作
        }
}else{
        $con->RollbackTrans ();                                                  //执行回滚操作
}
$rst->close ();                                                                  //关闭连接
$conn->close ();
?>
```

（3）创建 access.php 文件，循环输出 Access 数据库中存储的数据。

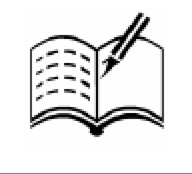
运行本示例，如图 9.17 所示，单击“数据转移”超链接，将 MySQL 数据库中的数据转移到 Access 数据库中，并且删除 MySQL 数据库中的原始数据。单击“查看 ACCESS”超链接，将查看到 Access 数据库中存储的数据，如图 9.18 所示。



图 9.17　数据转移



图 9.18　查看 Access 数据库中的数据



专家点评

事务是一个或一系列的查询，这些查询要么全部执行，要么全部不执行。其作用是在服务器发生错误或崩溃的情况下确保数据库的一致性。

一个事务被永久地写入到数据库中称事务提交，将状态重置到事务开始之前的状态称为事务回滚。事务具有以下 4 个特性。

- ☑ 原子性：指事务作为一个整体要么全部执行，要么全部不执行。
- ☑ 一致性：指一个事务必须能够使数据处于一致的状态。
- ☑ 孤立性：在事务完全完成之前，它们都是孤立的。
- ☑ 持续性：一旦写入数据库后，事务必须是永久的、持续的。

问题 208　ADODB 如何输出系统中执行的 SQL 语句？

问题阐述

ADODB 提供很多操作数据库的方法和函数，这其中不乏错误处理机制和程序调试的方法，那么如何输出系统中执行的 SQL 语句呢？

专家解答

输出系统中执行的 SQL 语句，应用的是 ADODB 中的 debug 变量。如果变量被设定为 True，那么当有输出操作时，同时也输出调试信息；如果启用 debug 变量，那么当错误发生时将自动调用 ErrorMsg()函数返回错误信息。

ErrorMsg()函数返回最后的状态或出错信息，即使没有错误发生，也会返回一个字串。所以，一般情况下，只有在发生错误时（返回 False）才调用该函数。

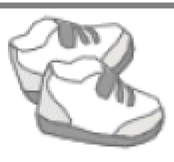
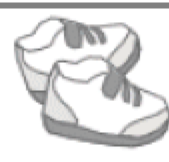
应用示例

在本示例中设置 debug 变量的值为 True，开启调试功能，输出程序中执行的 SQL 语句，如果 SQL 语句编写错误，那么将输出错误信息。其具体步骤如下：

（1）创建 conn 文件夹，编写 conn.php 文件，完成与 MySQL 数据库的连接。

（2）编写 index.php 文件。首先，包含数据库连接文件，设置 debug 变量的值为 True，开启调试功能。然后，定义 SQL 语句，循环输出 MySQL 数据库中的数据。最后，关闭连接、关闭数据库。其关键代码如下：

```
<?php
    include ("conn/conn.php");
```

```
            $conn -> debug = true;                                   //开启调试
            $sql = "select * from tb_use where id limit 5 ";         //创建 sql 语句
            $ret = $conn->execute ( $sql ) ;                         //执行查询语句
            while ( ! $ret->EOF ) {                                  //循环输出
    ?>
    <tr>
            <td height="20"><?php echo $ret->fields[0]; ?></td>
            <td height="20"><?php echo $ret->fields[1]; ?></td>
            <td><?php echo $ret->fields[3]; ?></td>
        </tr>
    <?php
            $ret->movenext ();                                       //指针下移
        }
        $ret->close ();                                              //关闭连接
        $conn->close ();
        ?>
```

运行结果如图 9.19 所示。

```
(mysql): select * from tb_use where id limit 5
1146: Table 'db_database13.tb_use' doesn't exist
ADOConnection._Execute(select * from tb_use where id limit 5 , false) % line 1015, file: adodb.inc.php
ADOConnection.Execute(select * from tb_use where id limit 5 ) % line    40, file: index.php
Fatal error: Call to a member function movenext() on a non-object in F:\AppServ\www\MR\13\013\index.php on line 57
```

图 9.19　输出 SQL 语句中的错误信息

专家点评

Debug 变量的调试功能适合在程序开发过程中应用，通过它可以及时发现 SQL 语句中的错误。但是在实际的应用环境中，就必须关闭这个调试功能，为了确保程序数据库的安全，如果存在 SQL 语句的错误，还是不要让浏览者看到为好，否则不但会影响到浏览者访问网站的心情，更重要的是容易暴露网站数据库的信息。

问题 209　ADODB 如何应用 errorMsg()函数返回错误信息？

问题阐述

ADODB 如何应用 errorMsg()函数返回错误信息？

专家解答

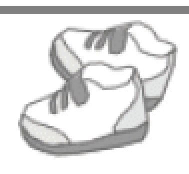
errorMsg()函数用于返回最后的状态或出错信息，即使没有错误发生，也会返回一个字串。所以，它多数情况下是与 die()函数联合使用的，在执行的 SQL 语句发生错误时（返回 False）才调用该函数。

应用示例

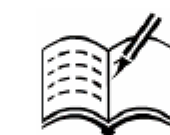
本示例应用 die()函数在执行的 SQL 语句发生错误时，调用 errorMsg()函数返回错误信息。其具体步骤如下：

（1）创建 conn 文件夹，编写 conn.php 文件，完成与 MySQL 数据库的连接。

（2）编写 index.php 文件。首先，包含数据库连接文件。然后，定义 SQL 语句，通过 execute()



函数执行 SQL 语句，应用 die()函数在 SQL 语句的执行发生错误时，调用 errorMsg()函数返回错误信息。最后，循环输出 MySQL 数据库中的数据。其关键代码如下：



```
<?php
        include ("conn/conn.php");
        $sql = "select * from tb_use where id limit 5 ";          //创建 sql 语句
        $ret = $conn -> execute($sql) or die($conn -> errorMsg());//执行查询语句，并返回错误信息
        while ( ! $ret->EOF ) {                                   //循环输出
?>
<tr>
        <td height="20"><?php echo $ret->fields[0]; ?></td>
        <td height="20"><?php echo $ret->fields[1]; ?></td>
        <td><?php echo $ret->fields[3]; ?></td>
    </tr>
<?php
        $ret->movenext ();                                        //指针下移
    }
    $ret->close ();                                               //关闭连接
    $conn->close ();
    ?>
```

运行结果如图 9.20 所示。

ErrorMsg返回错误信息

Table 'db_database13.tb_use' doesn't exist

ID　用户名　时间

图 9.20　返回错误信息

专家点评

errorMsg()函数返回最后的状态或出错信息，其特点是无论 SQL 语句是否出错它都会返回一个字符串，这就要求在应用 errorMsg()函数时必须注意。为避免在 SQL 语句正确的情况下返回字符串，所以 errorMsg()函数在 SQL 语句的执行发生错误时运用是最佳方案。

问题 210　ADODB 如何返回所有错误信息?

问题阐述

在 ADODB 中如何返回所有错误信息？

专家解答

ADODB 中提供的这个自定义的错误处理机制存储于 adodb-errorhandler.inc.php 文件中，通过它可以调用 PHP 函数 error_reporting 控制错误信息的输出样式。

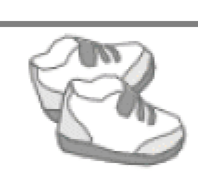
注意：

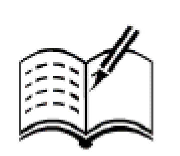
应用 ADODB 中提供的自定义错误处理机制，必须在程序连接数据库服务器之前载入 adodb-errorhandler.inc.php 文件。

应用示例

本示例运用 ADODB 类库中提供的自定义错误处理机制，调用 PHP 函数 error_reporting 设置错误级别为 E_ALL，显示所有的错误信息。其具体实现步骤如下：

（1）创建 conn 文件夹，编写 conn.php 文件。首先，设置 error_reporting()函数的值为 E_ALL，





Note

显示所有的错误信息。然后，载入 adodb-errorhandler.inc.php 文件。最后，完成与 MySQL 数据库的连接。其代码如下：

```
<?php
error_reporting(E_ALL);                                    //显示所有的错误信息
include_once ('../adodb5/adodb-errorhandler.inc.php');
include_once ('../adodb5/adodb.inc.php');                  //载入 adodb.ini.php 文件
$conn = ADONewConnection('mysql');                         //连接 MySQL 数据库
$conn -> PConnect('localhost','root','111','db_database'); //连接 db_database 库
//$ADODB_FETCH_MODE = ADODB_FETCH_ASSOC;                   //设置结果集以字段名称为索引
$conn -> execute('set names utf8');                        //设置编码
?>
```

（2）编写 index.php 文件。首先，包含数据库连接文件。然后，定义 SQL 语句，通过 execute()函数执行 SQL 语句。最后，循环输出 MySQL 数据库中的数据。其关键代码如下：

```
<?php
        include ("conn/conn.php");
        $sql = "select * from tb_use where id limit 5 ";   //创建 sql 语句
        $ret = $conn -> execute($sql);                     //执行查询语句
        while ( ! $ret->EOF ) {                            //循环输出
?>
<tr>
        <td height="20"><?php echo $ret->fields[0]; ?></td>
        <td height="20"><?php echo $ret->fields[1]; ?></td>
        <td><?php echo $ret->fields[3]; ?></td>
    </tr>
<?php
        $ret->movenext ();                                 //指针下移
    }
    $ret->close ();                                        //关闭连接
    $conn->close ();
    ?>
```

运行结果如图 9.21 所示。



图 9.21　返回所有错误信息

专家点评

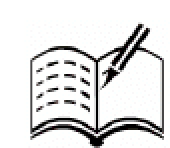
在本示例中，设置 error_reporting()函数的值为 E_ALL 返回所有的错误信息。如果将值设置为 0，则不会有任何错误信息输出。

问题 211　如何将错误信息存储于日志文件中？

问题阐述

通常只是应用 ADODB 类库中提供的自定义错误处理机制返回所有的错误信息，那么如何将错误信息存储于日志文件中呢？





专家解答

要想将错误信息存储于日志文件中，应用 ADODB 中提供的自定义错误处理机制，必须在程序连接数据库服务器之前载入 adodb-errorhandler.inc.php 文件。

将错误信息存储于指定的文件中，应用的是两个选择性常量：ADODB_ERROR_LOG_TYPE 和 ADODB_ERROR_LOG_DEST。

ADODB_ERROR_LOG_TYPE 常量设置错误记录的类型，可选值如下：

☑ 0：默认值，根据在 php.ini 文件中的 error_log 配置，错误被发送到服务器日志系统或文件。

☑ 1：错误被发送到 destination 参数中的地址。只有该类型使用 headers 参数。

☑ 2：通过 PHP debugging 连接来发送错误。该选项只在 PHP 3 中可用。

☑ 3：错误发送到文件目标字符串。

将 ADODB_ERROR_LOG_TYPE 的值设置为 3，即将错误信息存储于常量 ADODB_ERROR_LOG_ DEST 指定的文件中。

例如，指定 ADODB_ERROR_LOG_DEST 的值为 error/error.log，即将错误信息存储于项目根目录下 error 文件夹的 error.log 文件中。

应用示例

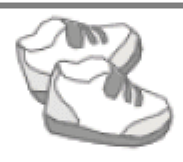
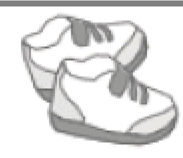
在本示例中，运用 ADODB 中的错误处理机制，不但返回错误信息，而且将错误信息存储到指定的文件中。其具体步骤如下：

（1）创建 conn 文件夹，编写 conn.php 文件。首先，定义两个常量设置程序中错误信息的存储位置。然后，载入 adodb-errorhandler.inc.php 文件。最后，完成与 MySQL 数据库的连接。其代码如下：

```
<?php
define('ADODB_ERROR_LOG_TYPE',3);                         //设置类型
define('ADODB_ERROR_LOG_DEST','error/error.log');         //设置存储路径
include_once ('../adodb5/adodb-errorhandler.inc.php');    //载入自定义错误处理文件
include_once ('../adodb5/adodb.inc.php');                 //载入 adodb.ini.php 文件
$conn = ADONewConnection('mysql');                        //连接 MySQL 数据库
$conn -> PConnect('localhost','root','111','db_database'); //连接 db_database 库
$conn -> execute('set names utf8');                       //设置编码
?>
```

（2）编写 index.php 文件。首先，包含数据库连接文件。然后，定义 SQL 语句，通过 execute()函数执行 SQL 语句。最后，循环输出 MySQL 数据库中的数据。其关键代码如下：

```
<?php
        include ("conn/conn.php");
        $sql = "select * from tb_use where id limit 5 ";    //创建 sql 语句
        $ret = $conn -> execute($sql);                      //执行查询语句
        while ( ! $ret->EOF ) {                             //循环输出
?>
<tr>
        <td height="20"><?php echo $ret->fields[0]; ?></td>
        <td height="20"><?php echo $ret->fields[1]; ?></td>
        <td><?php echo $ret->fields[3]; ?></td>
    </tr>
<?php
```

```
        $ret->movenext ();                              //指针下移
    }
    $ret->close ();                                     //关闭连接
    $conn->close ();
    ?>
```


Note

在本示例中将返回的错误信息存储于根目录下的 error 文件夹中，error.log 中的错误信息如图 9.22 所示。



图 9.22　通过日志文件存储错误信息

问题 212　如何封装 ADODB 操作数据库类?

问题阐述

既然可以将 ADODB 连接数据库的方法封装到类中，那么如何将 ADODB 操作数据库的方法封装到类中？

专家解答

ADODB 操作数据库类是 AdminDB，在操作数据库的 ExecSQL()方法中关键是如下几个函数的运用。

（1）strtolower()函数，将字符串转换为小写字母。其语法如下：

```
string strtolower(string str)
```

运用本函数的目的是将 SQL 语句中的字符串都转换为小写。

（2）substr()函数，从指定的字符串中按照指定的位置截取一定长度的字符。其语法如下：

```
string substr(string str,int start,int length)
```

参数说明：

❶ str：指定本截取的字符串。

❷ start：指定开始截取字符串的位置。如果参数 start 为负数，则从字符串的末尾开始截取。

❸ length：可选参数。指定截取字符串的长度。如果 length 为负数，则表示取倒数第 length 个字符。

技巧：

本函数中参数 start 的指定位置是从 0 开始计算的，即字符串中的第一个字符表示为 0。

运用 substr()函数对 SQL 语句中前 6 个字符串进行截取，其目的是获取 SQL 语句的类型（insert、select 或 update），判断其执行的是什么操作。

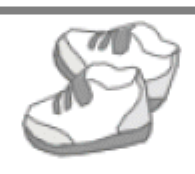
（3）trim()函数，删除字符串中首尾的空白或其他字符。

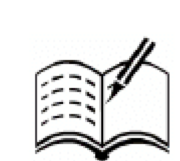
（4）execute()函数，ADODB 中执行 SQL 语句的方法，并返回一个结果集（ADORecordSet 对象），失败则返回 False。其语法如下：

```
execute($sql[,$inputarr=false])
```

参数说明：

❶ $sql：指定要执行的 SQL 语句。





❷ $inputarr：设置传入的结合变量，如果没有设置$inputarr，则$sql 为普通的 SQL 语句，那么 execute 的格式如下：

```
$connect->execute('select * from tb_user where id = 1')
```

如果设置了$inputarr，那么 execute 的格式如下：

```
$connect -> execute('select * from tb_user where id = ?',array($vl))
```

（5）GetRows()函数是 GetArray()的同义函数，是为了与 Microsoft ADO 相容才有的。其语法如下：

```
GetRows ([$number_of_rows])
```

返回从当前指针指向的记录开始，到（$number_of_rows - 1）行的全部记录的数组。参数$number_of_rows 是指定的记录行。如果没有给出，则一直到 EOF 才停止。

应用示例

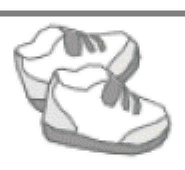
封装数据库操作类 AdminDB，同时应用数据库连接类 ConnDB 实现与 MySQL 数据库 db_database 的连接，应用 AdminDB 操作 MySQL 数据库，循环读取数据库中的数据。其具体步骤如下：

（1）创建 found.database.php 文件，定义数据库操作类 AdminDB，定义 ExecSQL()方法，根据传递的 SQL 语句执行对数据库的操作。在 ExecSQL()方法中，应用 substr()函数获取参数传递的 SQL 语句的类型，然后根据 SQL 语句的类型进行判断，如果是 select 查询语句，则执行 GetRow()方法，获取查询结果；如果是 update、insert 或 delete 语句，则直接返回 True 或 False。其代码如下：

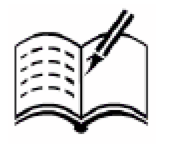
```
class AdminDB{
    function ExecSQL($sqlstr,$conn){   //定义方法，参数为 SQl 语句和连接数据库返回的对象
        $sqltype=strtolower(substr(trim($sqlstr),0,6));//截取 SQL 中的前 6 个字符串，并转换成小写
        $rs=$conn->Execute($sqlstr);              //执行 SQL 语句
        if($sqltype=="select"){                   //判断如果 SQL 语句的类型为 SELECT
            $array=$rs->GetRows();                //执行该语句，获取查询结果
            if(count($array)==0 || $rs==false)    //判断语句是否执行成功
                return false;                     //如果查询结果为 0 或执行失败，则返回 False
            else
                return $array;                    //否则返回查询结果的数组
        }elseif ($sqltype=="update" || $sqltype=="insert" || $sqltype=="delete"){
            //判断如果 SQL 语句类型不为 SELECT 则执行如下操作
            if($rs)
                return true;                      //执行成功返回 True
            else
                return false;                     //否则返回 False
        }
    }
}
```

（2）创建 connect.database.php 文件，实例化数据库连接类 ConnDB 和数据库操作类 AdminDB。其关键代码如下：

```
<?php
require("found.database.php");                                  //包含类文件
```



```
$connobj=new ConnDB("mysql","localhost","root","111","db_database",false); //数据库连接类实例化
$conn=$connobj->GetConnId();                                    //返回连接标识
$admindb=new AdminDB();                                         //数据库操作类实例化
?>
```



Note

（3）创建 index.php 文件，包含数据库连接类和数据库操作类的实例化文件，定义 SQL 语句，调用数据库操作类中的 ExecSQL()方法执行 SQL 查询语句，最后通过 for 语句循环 ExecSQL()方法返回的查询结果。其关键代码如下：

```
<?php
include_once 'conn/connect.database.php';          //包含数据库连接和操作类的实例化文件
$sqlstr = 'select * from tb_bccd';                //定义 SQL 语句
$array = $admindb->ExecSQL($sqlstr,$conn);        //调用数据库操作类中的方法，执行查询操作
for($i=0;$i<count($array);$i++){                  //通过 for 循环输出数据库中的数据
?>
  <tr>
    <td align="center" bgcolor="#FFFFFF"><?php echo $array[$i][0];?></td>
    <td align="center" bgcolor="#FFFFFF"><?php echo $array[$i][1];?></td>
    <td height="22" align="center" bgcolor="#FFFFFF"><?php echo $array[$i][2];?></td>
    <td align="center" bgcolor="#FFFFFF"><?php echo $array[$i][3];?></td>
  </tr>
  <?php
  }
?>
```

运行本示例，循环输出数据库中的数据，其运行结果如图 9.23 所示。



ID	名称	版本	时间
1	VC编程词典	标准版	2010-07-29
2	VB编程词典	标准版	2010-07-29
3	JAVA编程词典	标准版	2010-07-29

图 9.23　ADODB 操作数据库类

专家点评

SQL 语句的类型分为查询（select）、添加（insert）、更新（update）和删除（delete）4 种。这 4 个英文单词都有一个共同点，都是 6 个英文字母。我们就是根据它们的这个共同点对 SQL 语句的类型进行判断的，这也就是应用 substr()函数对 SQL 语句的前 6 个字符串进行截取的原因。

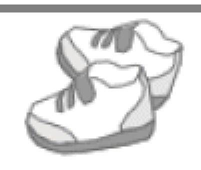
问题 213　如何封装 ADODB 分页类？

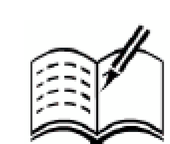
问题阐述

通过 ADODB 操作数据库中的数据时，常常会需要对输出的数据进行分页输出，如果在一个程序中需要多次应用分页输出，则可以将分页方法封装到类中，这样就可以在需要时直接调用，不必重复编写。那么如何封装 ADODB 分页类，使之在需要应用分页类时直接调用即可？

专家解答

在 ADODB 分页类 SepPage()中主要整合了下面几个 ADODB 中的函数。





（1）PageExecute()函数，分页功能函数，其语法如下：

```
PageExecute($sql, $nrows, $page);
```

参数说明：

❶ $sql：SQL 查询语句。

❷ $nrows：每页显示的记录数。

❸ $page：保存当前页数，默认为 1。

（2）AbsolutePage()函数，返回当前页的页数。其语法如下：

```
AbsolutePage($page=-1)
```

要与 PageExecute()函数配合使用。

（3）AtFirstPage()函数，如果当前页是第一页，则返回 True。其语法如下：

```
AtFirstPage($status='')
```

要与 PageExecute()函数配合使用，判断当前是否是第一页。

（4）AtLastPage()函数，判断当前页是否是最后一页。其语法如下：

```
AtLastPage($status='');
```

同样要与 PageExecute()函数配合使用，判断当前页是否是最后一页，如果是则返回 True。

应用示例

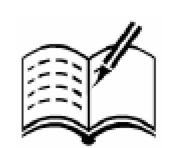
本示例封装一个 ADODB 分页类，并且运用 ADODB 分页类输出 MySQL 数据库 db_database 中 tb_user 数据表中的数据。其具体步骤如下：

（1）创建 found.database.php 文件，定义 SepPage 分页类及 ShowData()方法，读取数据库中存储的数据，并且将查询结果返回到数组中；定义 ShowPage()方法，创建分页超链接，完成从数据库中读取数据的分页输出。SepPage 分页类的关键代码如下：

```
class SepPage{
    var $rs;
    var $pagesize;
    var $nowpage;
    var $array;
    var $conn;
    var $sqlstr;
    function ShowData($sqlstr,$conn,$pagesize,$nowpage){      //定义方法
        if(!isset($nowpage) || $nowpage=="")                   //判断变量值是否为空
            $this->nowpage=1;                                  //定义每页起始页
        else
            $this->nowpage=$nowpage;
        $this->pagesize=$pagesize;                             //定义每页输出的记录数
        $this->conn=$conn;                                     //连接数据库返回的标识
        $this->sqlstr=$sqlstr;                                 //执行的查询语句
        $this->rs=$this->conn->PageExecute($this->sqlstr,$this->pagesize,$this->nowpage);
        @$this->array=$this->rs->GetRows();                    //获取记录数
            if(count($this->array)==0 || $this->rs==false)
                return false;
            else
                return $this->array;
```



```
    }
    //省略了 ShowPage()方法中的内容
}
```



（2）创建 connect.database.php 文件，实例化数据库连接类 ConnDB、数据库操作类 AdminDB 和分页类 SepPage。其关键代码如下：

```
<?php
require("found.database.php");                                      //包含类文件
$connobj=new ConnDB("mysql","localhost","root","111","db_database",false); //数据库连接类实例化
$conn=$connobj->GetConnId();
$admindb=new AdminDB();                                             //数据库操作类实例化
$seppage=new SepPage();                                             //分页类实例化
?>
```

（3）创建 index.php 文件，包含 connect.database.php 文件，调用分页类中的 ShowData()方法，完成数据的分页输出；调用分页类中的 ShowPage()方法输出分页的超链接。其关键代码如下：

```
<?php
include_once 'conn/connect.database.php';                    //调用类中方法
$array=$seppage->ShowData("select * from tb_user",$conn,10,$_GET["page"]);
                                                            //分页读取数据库中的数据
for($i=0;$i<count($array);$i++){                            //循环输出数据库中的数据
?>
<tr>
<td height="20" align="center" bgcolor="#FFFFFF"><?php echo $array[$i][0];?></td>
    <td align="center" bgcolor="#FFFFFF"><?php echo $array[$i][1];?></td>
    <td align="center" bgcolor="#FFFFFF"><?php echo $array[$i][2];?></td>
</tr>
<?php
}
?>
<tr>
    <td  height="35"  ><?php  echo  $seppage->ShowPage(" 用 户 "," 个 "," 参 数  1"," 参 数 2","a1");?></td>
</tr>
```

运行结果如图 9.24 所示。



图 9.24　ADODB 分页类的应用

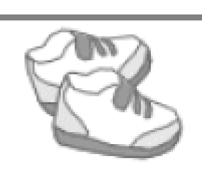
专家点评

在运用 ADODB 分页类进行数据的分页输出时，向 ShowData()方法中传递了 4 个参数：SQL 查询语句、数据库的连接标识、每页显示的记录数和当前页码；向 ShowPage()方法中传递了 5 个参数：数据的类型、数据的单位、超链接的参数一、参数二和超链接样式。



Ajax 技术

- ⏭ 问题 214　Ajax 不刷新页面提交表单数据
- ⏭ 问题 215　Ajax 无刷新验证用户名是否被占用
- ⏭ 问题 216　如何应用 Ajax 技术实现无刷新下拉列表?
- ⏭ 问题 217　如何实现 Ajax 无刷新分页?
- ⏭ 问题 218　Ajax 无刷新倒计时
- ⏭ 问题 219　如何让 Ajax 中的方法熟悉兼容各种不同的浏览器?
- ⏭ 问题 220　jQuery 特效复选框的制作方法
- ⏭ 问题 221　如何使用 jQuery 技术制作级联菜单?
- ⏭ 问题 222　如何通过 jQuery 生成网页特效?



问题 214　Ajax 不刷新页面提交表单数据



问题阐述

用户登录功能的原理是在客户端通过表单提交用户名和密码，将数据提交到服务器中，在服务器中完成对提交用户名和密码的验证，从而判断这个用户是否可以登录。这是该功能实现的基本原理，但是如果通过 Ajax 技术来实现该功能就不需要刷新页面，或者说不需要重新加载程序就可以完成用户名和密码的验证，从而减少了刷新页面的等待时间。那么如何实现 Ajax 不刷新页面提交表单数据呢？

专家解答

1. Ajax 工作原理

应用 Ajax 技术实现用户登录的功能，其中 Ajax 的工作原理如图 10.1 所示。

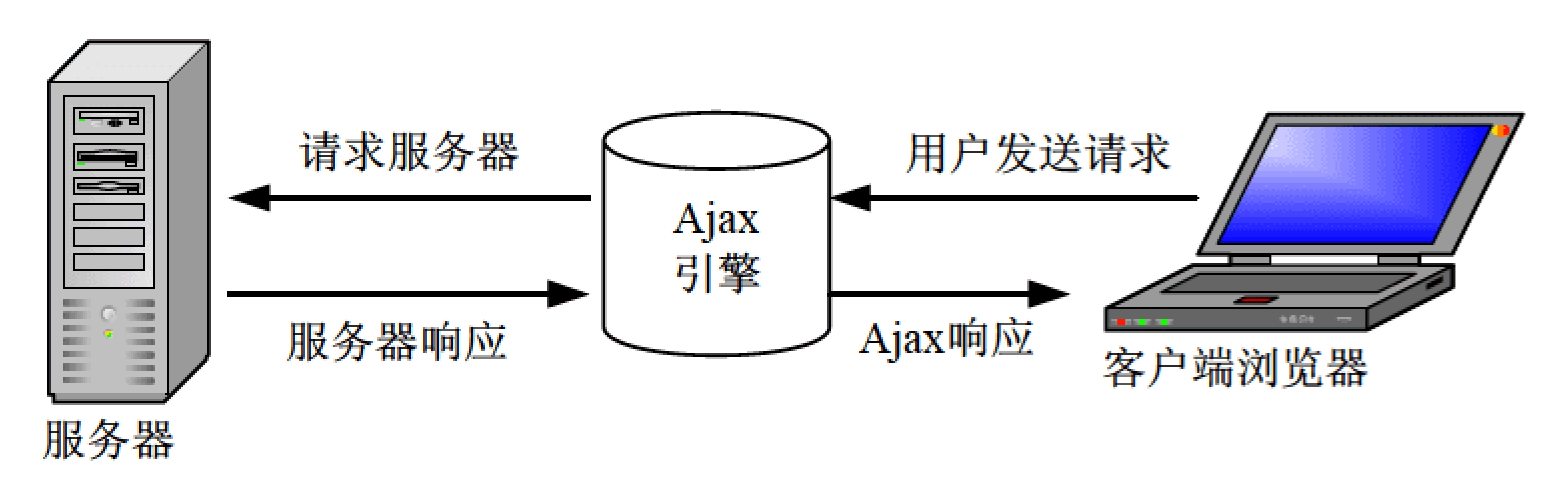


图 10.1　Ajax 工作原理

通过客户端浏览器发送请求（登录用户名和密码），通过 Ajax 直接请求服务器，服务器响应（检测登录用户名和密码），最终通过 Ajax 返回到客户端（输出验证结果）。

通过 Ajax 技术实现用户登录功能，其关键就是 XMLHttpRequest 的应用，它是一个具有应用程序接口的 JavaScript 对象，能够使用超文本传输协议（HTTP）连接一个服务器，是微软公司为了满足开发者的需要，于 1999 年在 IE 5.0 浏览器中率先推出的。

使用 XMLHttpRequest 对象，Ajax 可以像桌面应用程序一样只与服务器进行数据层面的交换，而不用每次都刷新页面，也不用每次都将数据处理的工作交给服务器来做，这样既减轻了服务器的负担，又加快了响应速度，缩短了用户等待的时间。

2. XMLHttpRequest 对象的常用方法

（1）open()方法

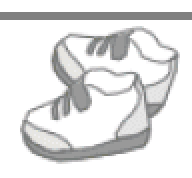
open()方法用于设置进行异步请求目标的 URL、请求方法以及其他参数信息，语法如下：

```
open("method","URL"[,asyncFlag[,"userName"[, "password"]]])
```

open()方法的参数说明如表 10.1 所示。

表 10.1　open()方法的参数说明

参　　数	说　　明
method	用于指定请求的类型，一般为get或post
URL	用于指定请求地址，可以使用绝对地址或者相对地址，并且可以传递查询字符串



续表

参　数	说　明
asyncFlag	可选参数，用于指定请求方式，同步请求为true，异步请求为false，默认情况下为true
userName	可选参数，用于指定请求用户名，没有时可省略
password	可选参数，用于指定请求密码，没有时可省略



（2）send()方法

send()方法用于向服务器发送请求。如果请求声明为异步，则该方法将立即返回，否则将等到接收到响应为止。语法格式如下：

```
send(content)
```

参数 content 用于指定发送的数据，可以是 DOM 对象的实例、输入流或字符串。如果没有参数需要传递，则可以将其设置为 null。

（3）setRequestHeader()方法

setRequestHeader()方法为请求的 HTTP 头设置值。语法格式如下：

```
setRequestHeader("label", "value")
```

参数 label 用于指定 HTTP 头；value 用于为指定的 HTTP 头设置值。

注意：

setRequestHeader()方法必须在调用 open()方法之后才能调用。

（4）abort()方法

abort()方法用于停止当前异步请求。

（5）getAllResponseHeaders()方法

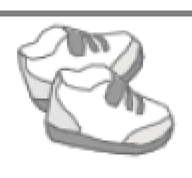
getAllResponseHeaders()方法用于以字符串形式返回完整的 HTTP 头信息，当存在参数时，表示以字符串形式返回由该参数指定的 HTTP 头信息。

3. XMLHttpRequest 对象的常用属性

XMLHttpRequest 对象的常用属性如表 10.2 所示。

表 10.2　XMLHttpRequest 对象的常用属性

属　性	说　明
onreadystatechange	每个状态改变时都会触发这个事件处理器，通常会调用一个JavaScript函数
readyState	请求的状态。有以下5个取值： 0 = 未初始化 1 = 正在加载 2 = 已加载 3 = 交互中 4 = 完成
responseText	服务器的响应，表示为字符串
responseXML	服务器的响应，表示为XML。这个对象可以解析为一个DOM对象





续表

属　　性	说　　明
status	返回服务器的HTTP状态码，如： 200="成功" 202="请求被接受，但尚未成功" 400="错误的请求" 404="文件未找到" 500="内部服务器错误"
statusText	返回HTTP状态码对应的文本

4. XMLHttpRequest 对象与服务器的交互

XMLHttpRequest 对象与服务器的交互分为以下 3 个步骤。

（1）初始化 XMLHttpRequest 对象。由于 XMLHttpRequest 不是一个 W3C 标准，所以对于不同的浏览器，初始化的方法也是不同的。

IE 浏览器把 XMLHttpRequest 实例化为一个 ActiveX 对象。具体方法为：

```
var http_request = new ActiveXObject("Msxml2.XMLHTTP");
```

或者

```
var http_request = new ActiveXObject("Microsoft.XMLHTTP");
```

上面语法中的 Msxml2.XMLHTTP 和 Microsoft.XMLHTTP 是针对 IE 浏览器的不同版本而进行设置的，是目前比较常用的两种。

Mozilla、Safari 等其他浏览器把 XMLHttpRequest 实例化为一个本地 JavaScript 对象。具体方法如下：

```
var http_request = new XMLHttpRequest();
```

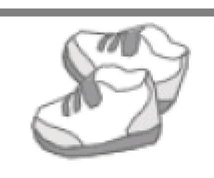
初始化 XMLHttpRequest 对象的代码如下：

```
var xmlHttp = false;        //定义 XMLHttpRequest 对象
try {                       //如果浏览器支持 XMLHttpRequest 对象，则创建 ActiveXObject 对象
        xmlHttp = new ActiveXObject("Msxml2.XMLHTTP");
} catch (e) {
        try {
                xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
        } catch (e2) {}
}
if (!xmlHttp && typeof XMLHttpRequest != "undefined") {
    try{
            xmlHttp = new XMLHttpRequest();
    }catch(e3){ xmlHttp = false;}
}
```

（2）设置请求状态和返回处理函数，语法格式如下：

```
xmlHttp.onreadystatechange=function_name;
```

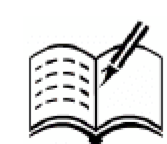
其中，xmlHttp 是 XMLHttpRequest 的对象；function_name 是用来处理请求状态和返回码的函数名（即回调函数）。



（3）发送 HTTP 请求，语法格式如下：

```
xmlHttp.open(send_method,url,flag);
```

其中，xmlHttp 是 XMLHttpRequest 的对象；send_method 是发送方法，可以是 GET()或 POST()其对应表单使用的方法；url 是页面要调用的地址；flag 是一个标记，如果为 True，则表示在等待被调用页面响应的时间内可以继续执行页面代码，反之为 False。



5. 应用示例

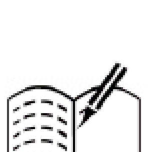
本示例应用 Ajax 技术实现用户的无刷新登录功能。其具体步骤如下：

（1）创建 conn.php 文件，连接 MySQL 数据库服务器，选择指定的数据库，并且设置数据库编码格式为 UTF-8。

（2）创建 index.php 文件，完成用户登录页面的设计。首先，嵌入 JavaScript 脚本文件 createxmlhttp.js，初始化 XMLHttpRequest 对象，嵌入 check.js 完成对用户提交用户名和密码的验证，并通过 open()方法调用 reg_chk.php 文件完成异步请求，同时将 Ajax 的响应通过 div 返回到客户端。check.js 的关键代码如下：

```
function showsimple(){                          //创建主控制函数
if(register.name.value==""){                    //判断用户名是否为空
        register.name.focus();
        return false;
    }
    if(register.pwd1.value==""){                //判断密码是否为空
        register.pwd1.focus();
        return false;
    }
    if(register.yzm.value==""){                 //判断验证码是否为空
        register.yzm.focus();
        return false;
    }
    if(register.yzm.value!=register.yzm2.value){   //判断输入的验证码是否正确
        alert("效验码输入错误！");
        register.yzm.focus();
        return false;
    }
    var username = document.getElementById("name").value;   //获取表单提交的用户名
    var password = document.getElementById("pwd1").value;   //获取密码
    var post_method="user="+username+"&pass="+password;     //构造 URL 参数
    xmlhttp.open("POST","reg_chk.php",true);                //调用指定的添加文件
    xmlhttp.setRequestHeader("Content-Type","application/x-www-form-urlencoded;");
                                                //设置请求头信息
    xmlhttp.onreadystatechange=StatHandler;     //判断 URL 调用的状态值并处理
    xmlhttp.send(post_method);                  //将数据发送给服务器
}
function StatHandler(){                         //定义处理函数
    if(xmlhttp.readyState==4 && xmlhttp.status==200){   //判断如果执行成功，则输出下面内容
        document.getElementById("webpage").innerHTML=xmlhttp.responseText;
                                                //将服务器返回的数据定义到 DIV 中
    }
}
```






```
function chkyzm(form){                                    //对验证码进行验证
    if(form.yzm.value==""){
        yzm1.innerHTML="<font color=#FF0000>请输入效验码！ </font>";
    }else if(form.yzm.value!=form.yzm2.value){
        yzm1.innerHTML="<font color=#FF0000>效验码输入错误!</font>";
    }else{
        yzm1.innerHTML="<font color=green>输入正确</font>";
    }
}
function yzm(form){                                       //生成验证码
    var num1=Math.round(Math.random()*10000000);
    var num=num1.toString().substr(0,4);
    document.write("<img name=codeimg src='yzm.php?num="+num+"'>");
    form.yzm2.value=num;
}
function code(form){                                      //重置验证码
    var num1=Math.round(Math.random()*10000000);
    var num=num1.toString().substr(0,4);
    document.codeimg.src="yzm.php?num="+num;
    form.yzm2.value=num;
}
```

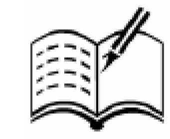
（3）在 reg_chk.php 文件中对 POST()方法提交的用户名和密码进行验证，并输出验证结果。其代码如下：

```
<?php
session_start();
header('Content-type: text/html;charset=utf-8');
include("conn.php");
$sql = "select * from tb_user where name='".$_POST['user']."' and password='".md5($_POST['pass'])."'";
                                                          //定义 SQL 语句
$mysql_query=mysql_query($sql,$conn);                     //执行 select 查询语句
$result=mysql_num_rows($mysql_query);
if($result<1){
        echo "登录失败，用户名或密码错误，请重新登录";
    }else{
        $_SESSION['member'] = $_POST['user'];
        echo  $_POST['user']."登录成功"."    "."<a  href='#'>由此进入主页
</a>";
    }
?>
```

（4）在 index.php 文件中创建表单，并提交用户名和密码，其中，通过 onBlur()事件对用户填写的数据进行验证，通过 onClick()事件调用 showsimple()方法完成用户登录的验证。最后，创建 div 标签，输出 Ajax 响应的结果。index.php 的关键代码如下：

```
<link rel="stylesheet" href="css/reg.css"/>
<script language="javascript" src="js/createxmlhttp.js"></script>
<script language="javascript" src="js/check.js"></script>
<table width="547" border="0" align="center" cellpadding="0" cellspacing="0">
<form id="register" name="register" method="post" >
    <tr>
```





```
                <td width="108" height="25"><div align="right">会员名：</div></td>
                  <td height="25" colspan="3"> 
                      <input id="name" name="name" type="text" onMouseOver="this.style.backgroundColor=
'#ffffff" onMouseOut="this.style.backgroundColor='#e8f4ff" /> <font color="red">*</font></td>
            </tr>
            <tr>
                  <td height="25"><div align="right">密 码：</div></td>
                  <td height="25" colspan="3"> 
                        <input id="pwd1" name="pwd1" type="password" onMouseOver="this.style
.backgroundColor='#ffffff"  onMouseOut="this.style.backgroundColor='#e8f4ff"/> <font  color="red">*
</font></td>
            </tr>
            <tr>
                  <td height="25"><div align="right">验证码：</div></td>
                  <td width="78" height="25"> 
                        <input  id="yzm"  type="text"  name="yzm"  size="8"  onBlur="javascript:chkyzm
(register)" onMouseOver="this.style.backgroundColor='#ffffff" onMouseOut="this.style.backgroundColor='#e8f4ff"/>
                        <input name="yzm2" type="hidden" value="" />
                </td>
                  <td width="63" align="center" valign="middle"><script>yzm(register)</script></td>
                  <td width="108"><a href="javascript:code(register)">  看不清</a></td>
                  <td height="25"><div id="yzm1"><font color="#999999">输入验证码</font></div></td>
                </tr>
                <tr>
                  <td height="50" colspan="5" align="center" valign="bottom"> 
                      <img src="images/03-11(2).jpg" width="91" height="31" onclick="showsimple();" />
                        <input  type="image"  name="imageField2"  src="images/03-11(3).jpg"  onclick=
"form.reset();return false;" /></td>
            </tr>
              </form>
            <tr>
                    <td height="50" colspan="5" align="center" valign="bottom"><div id="webpage" /></td>
                </tr>
        </table>
```

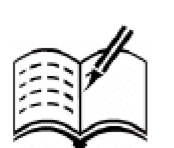
用户无刷新登录的运行结果如图 10.2 所示。



图 10.2　用户无刷新登录界面



问题 215　Ajax 无刷新验证用户名是否被占用



Note

问题阐述

在一些网站的注册功能中，为了避免出现同名的用户名，在注册时就对用户填写的用户名直接进行验证，如果用户名没有被占用，则可以使用，否则，将提示该用户名已被占用，那么用哪种方法对用户名验证最理想呢？

专家解答

用户名验证功能的实现采用 Ajax 技术是最理想的，因为其可以实现异步请求，无须重新加载页面即可直接获取验证结果。

通过 Ajax 技术实现无刷新验证用户名是否被占用，应用 XMLHttpRequest 对象实现异步请求，调用指定的文件完成用户名的验证操作。有关 Ajax 技术的详细讲解，读者可以参考问题 214 中的内容，这里不再赘述。

应用示例

在本示例中应用 Ajax 技术实现无刷新验证用户名是否被占用。其实现过程如下：

（1）创建数据库连接文件 conn.php，连接数据库服务器（用户名为 root，密码为 111），选择 db_database 数据库，设置数据库编码为 UTF-8。

（2）创建 js 文件夹，编写 JavaScript 脚本文件 createxmlhttp.js，初始化 XMLHttpRequest 对象；编写 check.js 完成对用户提交注册信息进行验证，并通过 open()方法调用 chkname.php 文件完成异步请求，验证用户填写的用户名是否被占用，并将 Ajax 的响应通过 div 返回到客户端，同时定义 yzm()和 code()两个方法，完成验证码的创建和无刷新更换。

check.js 的关键代码如下：

```
function chkname(form){
    if(form.name.value==""){                          //判断用户名表单元素值是否为空
        name1.innerHTML="<font color=#FF0000>请输入用户名！</font>";
                                                      //如果为空则输出该内容
    }else{
        var user = form.name.value;                   //否则获取用户名
        var url = "chkname.php?user="+user;           //定义 url
        xmlhttp.open("GET",url,true);                 //通过 get()方法执行
        xmlhttp.onreadystatechange = function(){      //回调函数
        if(xmlhttp.readyState == 4){                  //判断返回值
                var msg = xmlhttp.responseText;       //获取返回值
                if(msg == '2'){                       //判断如果返回值为 2
                    name1.innerHTML="<font color=#FF0000>用户名被占用！</font>";
                                                      //则说明用户名被占用
                    return false;
                }else if(msg == '1'){                 //判断如果返回值为 1
                    name1.innerHTML="<font color=green>恭喜您，可以注册!</font>";
                                                      //则说明用户名可用
                    form.c_name.value = "yes";
                }else{
```

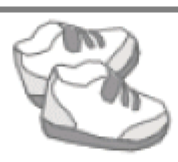
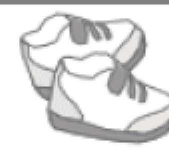





```
                        name1.innerHTML="<font color=green>"+msg+"</font>";
                                                        //输出可用用户名
                    }
                }
            }
            xmlhttp.send(null);                         //执行发送
        }
    }
    function chkinput(form){                            //验证表单中的元素是否为空，是否符合要求
        if(form.name.value==""){
            form.name.focus();
            return false;
        }
        if(form.c_name.value == "not"){
            form.name.select();
            return false;
        }
        //省略了部分代码
    }
    function chkpwd1(form){                             //无刷新验证注册信息，并返回验证结果
        if(form.pwd1.value==""){
            pwd11.innerHTML="<font color=#FF0000>密码格式错误！</font>";
        }else if(form.pwd1.value.length<6){
            pwd11.innerHTML="<font color=#FF0000>注册密码长度应大于 6 位！</font>";
        }else{
            pwd11.innerHTML="<font color=green>输入正确</font>";
        }
    }
    //省略了部分代码
    function yzm(form){                                 //创建验证码
        var num1=Math.round(Math.random()*10000000);
        var num=num1.toString().substr(0,4);
        document.write("<img name=codeimg src='yzm.php?num="+num+"'>");
        form.yzm2.value=num;
    }
    function code(form){                                //验证码的无刷新更换
        var num1=Math.round(Math.random()*10000000);
        var num=num1.toString().substr(0,4);
        document.codeimg.src="yzm.php?num="+num;
        form.yzm2.value=num;
    }
```

在 check.js 文件中调用 chkname.php 文件完成对用户名的验证，其传递参数是用户提交的用户名。调用 yzm.php 文件完成验证码的生成和无刷新更换，其传递参数是随机生成的验证码。chkname.php 的关键代码如下：

```
<?php
session_start();
include("conn.php");
$reback = '0';                                          //定义变量
$sql = "select * from tb_user where name='".$_GET['user']."'";   //定义 SQL 语句
$mysql_query=mysql_query($sql,$conn);                   //执行 select 查询语句
```

```
$result=mysql_num_rows($mysql_query);
if($result>0){
    $reback = '2';                    //则为变量赋值为 2
}else{
    $reback = '1';                    //如果查询结果不为真，则为变量赋值为 1
}
echo $reback;                         //输出变量值
?>
```

在 yzm.php 文件中，以生成的随机数为根本，通过 GD2 函数生成数字验证码，其代码如下：

```
<?php
srand((double)microtime()*1000000);                //生成随机数
$im=imagecreate(60,20);                            //创建画布
$black=imagecolorallocate($im,0,0,0);              //定义背景
$white=imagecolorallocate($im,255,255,255);        //定义背景
$gray=imagecolorallocate($im,200,200,200);         //定义背景
imagefill($im,0,0,$gray);                          //填充颜色
for($i=0;$i<4;$i++){                               //定义 4 位随机数
 $str=mt_rand(1,5);                                //定义随机字符所在位置的 Y 坐标
 $size=mt_rand(6,9);                               //定义随机字符的字体
 $authnum=substr($_GET[num],$i,1);                 //获取超链接中传递的验证码

imagestring($im,$size,(3+$i*15),$str,$authnum,imagecolorallocate($im,rand(0,130),rand(0,130),rand(0,130)));
}                                                  //水平输出字符串
for($i=0;$i<200;$i++){                             //执行 for 循环，为验证码添加模糊背景
  $randcolor=imagecolorallocate($im,rand(0,255),rand(0,255),rand(0,255)); //创建背景
  imagesetpixel($im,rand()%70,rand()%30,$randcolor);          //绘制单一元素
}
imagepng($im);                                                //生成 png 图像
imagedestroy($im);                                            //销毁图像
?>
```

（3）创建 index.php 文件，设计用户注册页面。首先通过 script 标签调用 js 脚本文件，然后创建 form 表单，并在该表单中通过 JavaScript 脚本事件完成对用户注册信息的验证，通过 div 标签输出验证结果，同时将注册信息提交到 reg_chk.php 文件中，将用户注册信息添加到数据库中。其关键代码如下：

```
<script language="javascript" src="js/createxmlhttp.js"></script>
<script language="javascript" src="js/check.js"></script>
<body onLoad="javascript:register.name.focus()">
<table width="547" border="0" align="center" cellpadding="0" cellspacing="0">
 <form id="register" name="register" action="reg_chk.php" method="post" onSubmit="return
chkinput(this)">
    <tr>
      <td width="108" height="25"><div align="right">用户名：</div></td>
      <td height="25" colspan="3"> 
          <input id="name" name="name" type="text" onBlur="javascript:chkname(register)"
onMouseOver="this.style.backgroundColor='#ffffff" onMouseOut="this.style.backgroundColor='#e8f4ff" />
          <input id="c_name" name="c_anme" type="hidden" value="not" > <font
color="red">*</font></td>
      <td height="25"><div id="name1"><font color="#999999">请输入用户名</font></div></td>
```



```
        </tr>
        <tr>
          <td height="25"><div align="right">验证码：</div></td>
          <td width="78" height="25"> 
            <input id="yzm" type="text" name="yzm" size="8" onBlur="javascript:chkyzm(register)"
onMouseOver="this.style.backgroundColor='#ffffff'" onMouseOut="this.style.backgroundColor='#e8f4ff'"/>
            <input name="yzm2" type="hidden" value="" /></td>
          <td width="63" align="center" valign="middle"><script>yzm(register)</script></td>
          <td width="108"><a href="javascript:code(register)">  看不清</a></td>
          <td height="25"><div id="yzm1"><font color="#999999">输入验证码</font></div></td>
        </tr>
        <tr>
          <td height="50" colspan="5" align="center" valign="bottom"> 
              <input type="image" name="imageField" src="images/03-11(2).jpg" />  
            <input type="image" name="imageField2" src="images/03-11(3).jpg" onclick="form.reset();
return false;" /></td>
          </tr>
      </form>
    </table>
```

（4）创建 reg_chk.php 文件，获取表单中提交的数据，并将用户注册信息添加到指定的数据表中。其运行结果如图 10.3 所示。

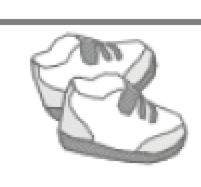


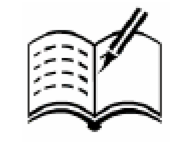
图 10.3 Ajax 无刷新验证用户名是否被占用

问题 216 如何应用 Ajax 技术实现无刷新下拉列表？

问题阐述

Ajax 技术可以应用到用户登录和用户注册的实现中，那么该技术能否在下拉列表中使用呢？





Note

专家解答

Ajax 技术是可以在下拉列表中使用的，如在下拉列表 A 中选择一个指定的值，当鼠标光标失去焦点并发生变化时，会在下拉列表 B 中输出一个与下拉列表 A 对应的值。其原理是根据下拉列表 A 中值的变化，下拉列表 B 中也同时生成一个对应的值。在这个过程中，下拉列表 A 中的值是从数据库中读取的，并且通过 onChange()事件调用 JavaScript 脚本函数，应用 Ajax 将下拉列表的值传递到指定的文件中，根据传递的值在指定的文件中生成下拉列表 B 的值，最后通过 Ajax 将下拉列表 B 的值返回到客户端。

应用示例

在本示例中，当用户选择左侧的“所属大类”时，随之右侧“所属小类”将出现大类中对应的小类。其运行结果如图 10.4 所示。



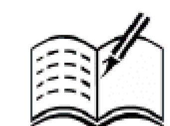
图 10.4　Ajax 实现无刷新的下拉列表

具体实现的过程如下：

（1）创建 index.php 文件，完成图书信息添加页面的设计。首先通过 script 标签调用 js 脚本文件，用于完成 Ajax 的异步请求和对表单元素值的验证。然后创建 form 表单，设置下拉列表完成图书类别输出，并通过 onChange 事件调用 js 文件中的函数，实现 Ajax 异步请求，完成下拉列表的关联。最后通过 onClick 事件调用 js 文件中的函数，对表单中的元素值进行判断，完成数据提交。其关键代码如下：

```
<script language="javascript" src="js/reg.js"></script>
<table width="575" border="0" align="center" cellpadding="-2" cellspacing="-2" bordercolordark="#FFFFFF">
<form action="" method="post" name="form1">
    <tr>
        <td height="27" align="right"> 所属大类：</td>
        <td width="31%" height="27"> 
        <select name="type" class="textarea" id="type" onChange="F_super(this.value)">
<?php
    $link = mysql_connect ( "localhost", "root", "root" );          //连接数据库服务器
    mysql_select_db ( "db_database", $link );                       //连接数据库
    //Ajax 中先用 encodeURIComponent 对要提交的中文进行编码
```

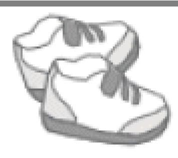




```
        $GB2312string = iconv ( 'UTF-8', 'gb2312//IGNORE', $RequestAjaxString );
        mysql_query ( "set names gb2312" );                    //设置数据库编码格式
        $sql = mysql_query ( "select * from tb_type group by type" );     //执行查询操作
        $result = mysql_fetch_object ( $sql );        //获取结果集中的一行记录
        do {                                          //循环输出查询结果，将结果作为下拉列表的值
            header ( 'Content-type: text/html;charset=GB2312' );   //指定发送数据的编码格式为 GB2312
    ?>
            <option value="<?php echo $result->type; ?>" selected><?php echo $result->type; ?> </option>
    <?php
        } while ( $result = mysql_fetch_object ( $sql ) );
    ?>
            </select>
            </td>
            <td width="13%" height="27" align="right"> 所属小类：</td>
            <td width="42%" height="27" id="subType"> </td>
        </tr>
        <tr>
            <td height="28" colspan="4" align="center">
                <input   name="Button"   type="button"   class="btn_grey"   value=" 保 存 "   onClick=
"mycheck();">
                <input name="Submit2" type="reset" class="btn_grey" value="重置">
            </td>
        </tr>
        </form>
    </table>
    <script language="javascript">F_super(form1.type.value);</script>
```

（2）创建 js 文件夹，编写 reg.js 脚本文件，定义 createRequest()函数，实现 Ajax 的异步请求；定义 alertContents()函数，作为回调函数，返回服务器的响应；定义 F_super()函数，调用 createRequest()函数，将 selSubType.php 文件和下拉列表的值作为参数；定义 mycheck()函数，对表单中的元素值进行判断，并执行提交操作。其关键代码如下：

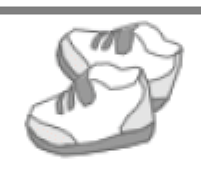
```
    var http_request = false;
    function createRequest(url) {                    //初始化对象并发出 XMLHttpRequest 请求
        http_request = false;
        if (window.XMLHttpRequest) {                 //Mozilla 或其他除 IE 以外的浏览器
            http_request = new XMLHttpRequest();
            if (http_request.overrideMimeType) {
                http_request.overrideMimeType("text/xml");
            }
        } else if (window.ActiveXObject) {           //IE 浏览器
            try {
                http_request = new ActiveXObject("Msxml2.XMLHTTP");
            } catch (e) {
                try {
                    http_request = new ActiveXObject("Microsoft.XMLHTTP");
                } catch (e) {}
            }
        }
        if (!http_request) {
```

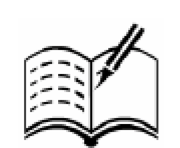



```
            alert("不能创建 XMLHTTP 实例!");
            return false;
        }
        http_request.onreadystatechange = alertContents;            //指定响应方法
        http_request.open("GET", url, true);                        //发出 HTTP 请求
        http_request.send(null);                                    //执行发送
    }
    function alertContents() {                                      //处理服务器返回的信息
        if (http_request.readyState == 4) {
            if (http_request.status == 200) {
                subType.innerHTML=http_request.responseText;  //将服务器返回值赋给指定 ID
            } else {
                alert('您请求的页面发现错误');
            }
        }
    }
    function F_super(val){
        createRequest("selSubType.php?type="+val+'&nocache='+new Date().getTime());
                                                                    //实现级联下拉列表
    }
    function mycheck(){
        if (form1.introduce.value==""){
            alert("请输入商品简介！ ");form1.introduce.focus();return;
        }
        form1.submit();
    }
```

（3）创建 selSubType.php 文件，根据 Ajax 异步请求中提交的下拉列表选项的值，从数据库中读取出对应的数据，并且将读取的数据作为下拉列表的值，重新生成一个下拉列表。其关键代码如下：

```
    <?php
        $link=mysql_connect("localhost","root","root");                 //连接数据库服务器
        mysql_select_db("db_database",$link);                           //连接数据库
        //Ajax 中先用 encodeURIComponent 对要提交的中文进行编码
        $GB2312string=iconv( 'UTF-8', 'gb2312//IGNORE',$RequestAjaxString);
        mysql_query("set names gb2312");                                //设置编码格式
        $type=$_GET[type];                                              //获取 Ajax 中传递的值
        $sql=mysql_query("select * from tb_type where type='$type'");   //执行查询操作
        $result=mysql_fetch_array($sql);                        //获取查询结果
        header('Content-type: text/html;charset=GB2312');       //指定发送数据的编码格式为 GB2312
    ?>
        <select name="typeID" class="textarea" id="typeID">
    <?php
        do{
    ?>
            <option value="<?php echo $result[subtype];?>"><?php echo $result[subtype];?></option>
    <?php
        }while($result=mysql_fetch_array($sql));
    ?>
        </select>
```





问题 217　如何实现 Ajax 无刷新分页？

问题阐述

有时人们在线看视频时，会看到下面有很多评论，当查看评论时，在执行翻页的操作后，视频文件又被重新打开，由于网络原因，当再次打开视频时发现其播放速度会变得很慢，那么如何才能实现在进行翻页操作时，不影响视频的播放呢？最理想的解决方案就是应用 Ajax 技术实现无刷新分页。

专家解答

所谓无刷新分页，就是在进行翻页的过程中不需要重新加载页面，而是在当前页面中完成翻页的操作。

通过 Ajax 技术实现无刷新分页功能的关键是 XMLHttpRequest 的应用，它可以像桌面应用程序一样只与服务器进行数据层面的交互，而不用每次都刷新页面，也不用每次都将数据处理的工作交给服务器来做，这样既减轻了服务器的负担，又加快了响应速度，缩短了用户等待的时间。在实现 Ajax 无刷新分页时将应用到如下所示的 Ajax 技术。

（1）open()方法

open()方法的具体内容请参考问题 214。

（2）send()方法

send()方法的具体内容请参考问题 214。

（3）XMLHttpRequest 对象的常用属性

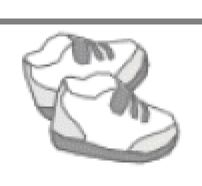
XMLHttpRequest 对象的常用属性如表 10.2 所示

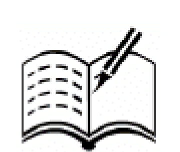
应用示例

本示例在播放.flv 文件的同时完成数据的无刷新分页输出，其运行结果如图 10.5 所示。



图 10.5　Ajax 无刷新分页





在本示例中实现的是超长文本中数据的分页读取，为了体现 Ajax 无刷新的特点，在页面中还插入了一个.flv 格式的视频。其具体实现步骤如下：

（1）创建 index.php 页面。包括 3 个方面的内容：一是嵌入.flv 格式的视频文件；二是完成超长文本的分页输出；三是通过 script 标签嵌入 js 脚本文件，定义 div 标签，为无刷新分页做准备。

第一方面，嵌入.flv 格式的视频文件，其代码如下：

```
<div style="width:520px; height=185 margin:0px; border:solid 5px #fcedda;">
    <div id="flash5" class="right">Flvplayer 播放器样式 1</div>
</div>
<script type="text/javascript" src="js/swfobject.js" ></script><!-- 载入 js 脚本文件 -->
<script type="text/javascript">
    var s5 = new SWFObject("FlvPlayerV2009.swf","mediaplayer","520","240","8");
    s5.addParam("allowfullscreen","true");
    s5.addVariable("width","520");                    //设置宽度
    s5.addVariable("height","240");                   //设置高度
    s5.addVariable("image","images/flash3.jpg");      //设置背景
    s5.addVariable("file","player.flv");              //指定播放文件
    s5.addVariable("backcolor","0xff8c00");           //设置背景颜色
        s5.addVariable("frontcolor","0xE2F0FE");      //设置字体颜色
    s5.write("flash5");                               //将内容写入 flash5
</script>
```

第二方面，超长文本的分页输出。该技术和其他分页技术相同，但是有一点必须注意：在无刷新分页的超链接中，href 的值不是一个具体的地址，而是“#”。无刷新分页中通过 onClick 事件调用自定义的方法进行异步请求，而调用另外一个文件执行页面的跳转。其关键代码如下：

```
<a href="#" onClick='return no_refurbish_pagination("index_ok.php?page=1")'>首页</a> 
    <a href="#" onClick='return no_refurbish_pagination("index_ok.php?page=<?php echo $_GET['page']-1;?>")'>
上一页</a>
    <a href="#" onClick='return no_refurbish_pagination("index_ok.php?page=<?php echo $_GET['page']+1;?>")'>
下一页</a>
    <a href="#" onClick='return no_refurbish_pagination("index_ok.php?page=<?php echo $page_count;?>")'>
尾页</a>
```

这里调用的是 no_refurbish_pagination()方法执行异步请求，请求的文件是 index_ok.php，通过该文件执行分页跳转。

第三方面，为 Ajax 无刷新做准备，嵌入 js 脚本文件 discuss_js.js，定义 div 标签，设置标签的 ID 为 synopsis，div 标签中的内容是在分页中重新载入的内容。

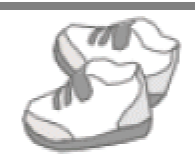
技巧：

在<div id="synopsis">…</div>中存储的必须是一个完整的 table 表格，否则是无法重新载入的。在这个 div 中存储的内容就是异步请求文件 index_ok.php 中的内容。

（2）创建 index_ok.php 文件，定义异步请求文件的内容，也就是重新保存 index.php 中<div id="synopsis">包含的内容。这个 index_ok.php 中存储的是整个超长文本的分页读取方法。

技巧：

在异步请求文件 index_ok.php 中需要设置页面的编码格式，否则在重新载入时可能会出现乱码的问题。





（3）创建 discuss_js.js 脚本文件，定义 no_refurbish_pagination()方法，执行异步请求操作。其代码如下：

```
var xmlHttp = false;
try {
    xmlHttp = new ActiveXObject("Msxml2.XMLHTTP");
} catch (e) {
    try {
        xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
      } catch (e2) {}
}
if (!xmlHttp && typeof XMLHttpRequest != "undefined") {
    try{
        xmlHttp = new XMLHttpRequest();
    }catch(e3){ xmlHttp = false;}
}
//使用 XMLHttpRequest 对象创建异步 HTTP 请求
function no_refurbish_pagination(url){            //创建自定义函数，获取传递的参数
    xmlHttp.open('get',url,true);                 //根据传递的参数，通过 get()方法执行异步请求
    xmlHttp.onreadystatechange = function(){
        if(xmlHttp.readystate == 4 && xmlHttp.status == 200){      //返回结果
            document.getElementById("synopsis").innerHTML = xmlHttp.responseText;
        }
    }
    xmlHttp.send(null);
}
```

专家点评

以上就是 Ajax 无刷新分页的步骤，分析其原理，就是在原有分页输出数据的基础上又定义了一个相同的内容，而该内容存储于 B 文件中。所谓的无刷新就是利用了 Ajax 可以直接向服务器发送请求的特点直接请求 B 文件，并将返回数据在原有文件中输出。

在实现 Ajax 无刷新分页时，还需要注意以下几点。

1. 超链接 href 值的设置

在通过 Ajax 实现无刷新分页的过程中，在创建分页超链接时，href 的值设置为“#”，通过 onClick 事件调用 no_refurbish_pagination()方法完成分页操作。

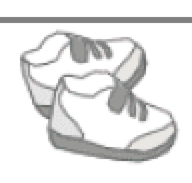
这里必须要注意 href 的值，如果此时将 href 的值设置为空（“ ”），那么就不能实现页面跳转，因为其链接的是当前页。

2. no_refurbish_pagination()方法中的参数

onClick 事件调用 no_refurbish_pagination()方法完成分页操作，在 no_refurbish_pagination()方法中传递了一个参数，即异步请求的文件和相应的参数值。在 no_refurbish_pagination()方法中定义该参数时，如果该参数中存在 PHP 脚本，那么必须要使用双引号或单引号进行定义，其正确格式如下：

```
<a href="#" onClick='return no_refurbish_pagination("index_ok.php?page=<?php echo $_GET['page']+1;?>")'>
下一页</a>
```

如果 onClick 事件使用的是单引号，那么在定义 no_refurbish_pagination()方法的参数时就





要使用双引号，格式如下：

```
<a href="#" onClick="return no_refurbish_pagination('index_ok.php?page=<?php echo $page_count;?> ') ">
尾页</a>
```

如果 onClick 事件使用的是双引号，那么在定义 no_refurbish_pagination()方法的参数时就要使用单引号。

如果不使用双引号或单引号，那么将输出如图 10.6 所示的错误信息。

如果 onClick 事件使用单引号，而在定义 no_refurbish_pagination()方法的参数时也使用单引号，或者 onClick 事件使用双引号，而在定义 no_refurbish_pagination()方法的参数时也使用双引号，那么将输出如图 10.7 所示的错误信息。

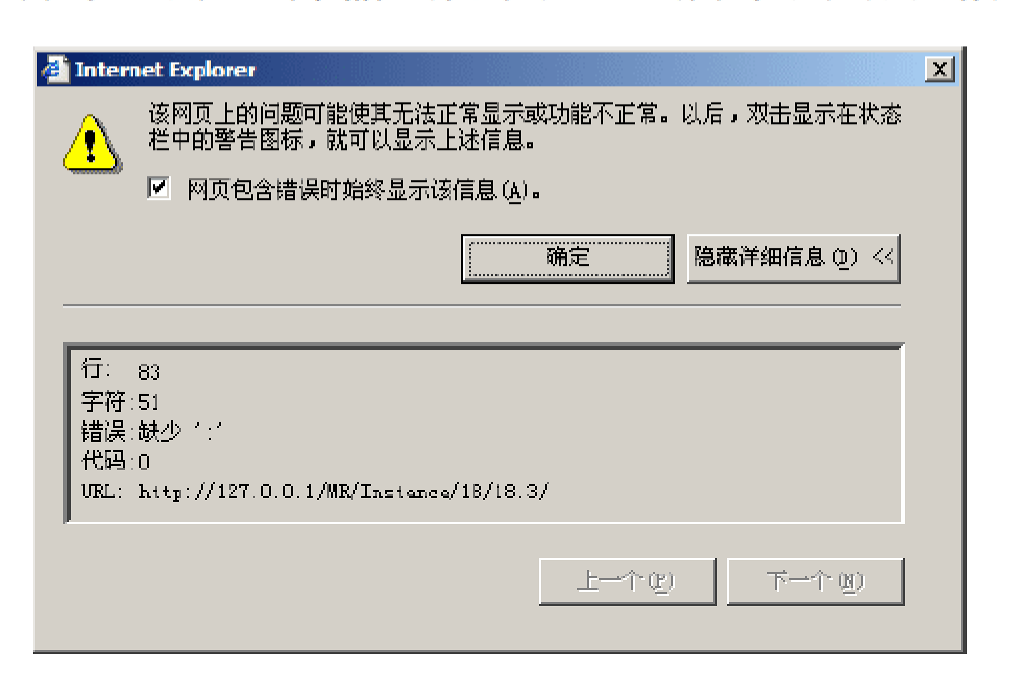


图 10.6　错误提示信息（一）

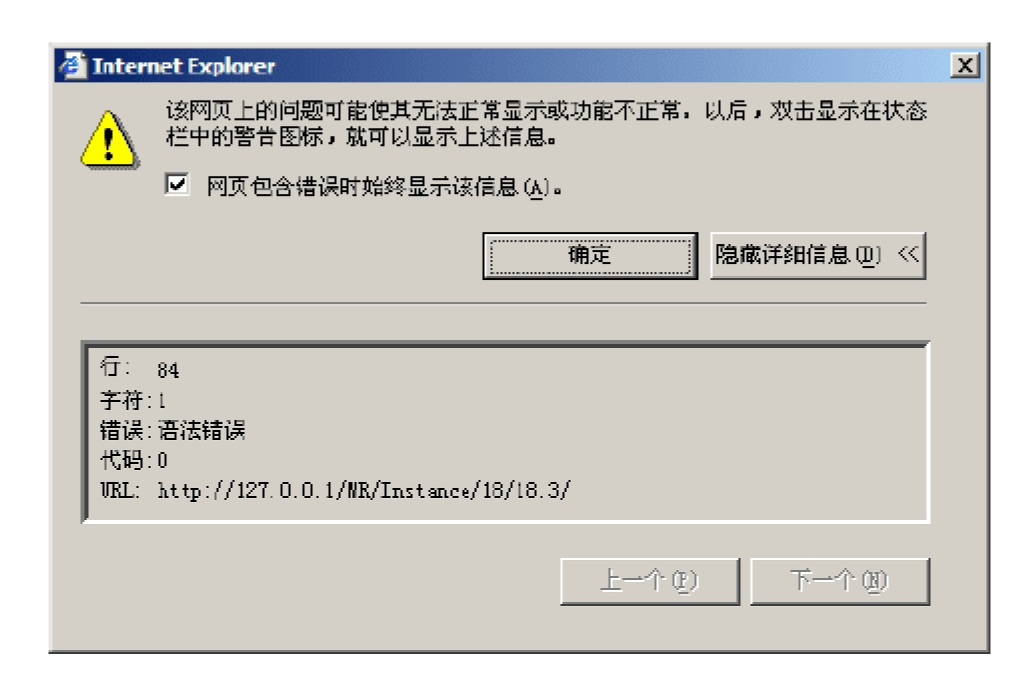


图 10.7　错误提示信息（二）

问题 218　Ajax 无刷新倒计时

问题阐述

在网页中实现倒计时非常普遍，其原理也很简单，就是用指定日期的时间戳减去当前日期的时间戳，得到的就是距离目标日期的期限。但是当倒计时的时间以分秒进行计算时，这个问题似乎就变得有点“意思”了，因为毕竟不能通过手动来不停地刷新网页，达到刷新倒计时时间的目的，那么就必须实现倒计时时间的无刷新输出，这就是这里要解决的问题——Ajax 无刷新倒计时。

专家解答

既然要通过 Ajax 实现无刷新倒计时，那么首先必须初始化 XMLHttpRequest 对象，并且保证其对不同浏览器的支持。初始化的方法如下：

```
var xmlHttp = false;
try {
        xmlHttp = new ActiveXObject("Msxml2.XMLHTTP");
} catch (e) {
        try {
                xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
        } catch (e2) {}
}

if (!xmlHttp && typeof XMLHttpRequest != "undefined") {
```



```
        try
        {
                xmlHttp = new XMLHttpRequest();
        }catch(e3){ xmlHttp = false;}
    }
```



然后调用 XMLHttpRequest 对象中的 open()方法，执行异步请求。open()方法用于设置进行异步请求目标的 URL、请求方法以及其他参数信息。具体语法如下：

```
open("method","URL"[,asyncFlag[,"userName"[, "password"]]])
```

在上面的语法中，method 用于指定请求的类型，一般为 get 或 post；URL 用于指定请求的地址，可以使用绝对地址或相对地址，并且可以传递查询字符串；asyncFlag 为可选参数，用于指定请求方式，同步请求为 true，异步请求为 false，默认情况下为 true；userName 为可选参数，用于指定请求用户名，没有时可省略；password 为可选参数，用于指定请求密码，没有时可省略。

接着通过 XMLHttpRequest 对象中的 onreadystatechange 属性调用 JavaScript 脚本函数，通过 responseText 属性获取服务器响应返回的字符串。XMLHttpRequest 对象的常用属性如表 10.2 所示。

最后通过 XMLHttpRequest 对象中的 send()方法向服务器发送请求。如果请求声明为异步，则该方法将立即返回，否则将等到接收到响应为止。其语法如下：

```
send(content)
```

在上面的语法中，content 用于指定发送的数据，可以是 DOM 对象的实例、输入流或字符串。如果没有参数需要传递，则可以设置为 null。

这是无刷新倒计时中 Ajax 部分应用到的技术、方法。另外，还有与 PHP 结合的内容，即通过 open()方法异步请求的文件，在请求的文件中，计算出倒计时的用时及剩余时间，最终通过 responseText 属性获取返回的时间字符串，并且将其赋给指定的 div 标签，在页面中输出。

在计算倒计时的用时和剩余时间时应用 mktime()函数获取系统的当前时间戳、倒计时开始的时间戳以及倒计时结束的时间戳，只需对这 3 个值做减法运算，即可获取倒计时的用时和剩余时间。其关键代码如下：

```
<?php
session_start();                              //初始化 SESSION 变量
header('Content-Type: text/xml');             //定义页面文件格式
$dates=$_SESSION['dates'];                    //获取计时开始时间
$dates2=mktime();                             //获取当前时间
$dates3=$dates2-$dates;                       //计算倒计时的用时
echo date("i:s",$dates3);                     //格式化输出用时
?>
```

应用示例

应用 Ajax 实现无刷新倒计时，设计倒计时的时长为 20 分钟，当倒计时结束后给出提示信息。首先，通过 mktime()函数获取一个倒计时开始的时间戳，将这个值提交到倒计时输出页面，存储到 SESSION 变量中，在输出页面中定义两个 JavaScript 脚本函数 ShowTime()和 sparetime()，通过异步请求调用 showtime.php 和 sparetime.php 这两个文件，获取到倒计时的用时和剩余时间，最后将获取的时间在 div 标签中输出。ShowTime()方法的代码如下：




```
<script type="text/javascript">
timer = window.setInterval("ShowTime()",1000);
function ShowTime(){
    xmlHttp.open("post","showtime.php", true);
    xmlHttp.onreadystatechange = function(){
        if(xmlHttp.readyState == 4){
            tet = xmlHttp.responseText;
            document.getElementById("show_time").innerHTML = tet;
        }
    }
    xmlHttp.send(null);
}
</script>
<div class="STYLE1" id="show_time"></div>
```

showtime.php 文件的代码如上面所示。

运行结果如图 10.8 所示。



图 10.8　Ajax 无刷新倒计时

专家点评

无刷新倒计时运用 Ajax 在不需要重新载入整个页面的情况下，XML-HttpRequest 对象就可以向服务器发送请求并得到服务器响应的特性，实现倒计时时间在页面中无刷新地更新输出。

在应用 Ajax 时，由于 Ajax 不支持多种字符集，它默认的字符集是 UTF-8，所以在应用 Ajax 技术的程序中应及时进行编码转换，避免在输出中文字符串时出现乱码。最简单有效的方法就是程序统一使用 UTF-8 编码，这时就不会涉及编码转换或者乱码的问题了。

问题 219　如何让 Ajax 中的方法熟悉兼容各种不同的浏览器？

问题阐述

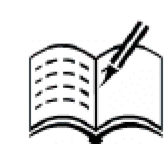
很多时候用户在 IE 浏览器中使用 Ajax 实现的功能是正常的，但是当在 Firefox（火狐）等其他浏览器中运行相同的程序时，却得不到同样的效果，甚至有些功能不能正常运行。这就是由于 Ajax 在不同的浏览器中使用的方法不兼容导致的。

专家解答

解决 Ajax 在不同浏览器中不兼容的问题，其方法就是自定义一个函数，判断当使用不同的浏览器时，返回不同的内容。其定义的方法如下：

```
function getOs(){
   var OsObject = "";
   if(navigator.userAgent.indexOf("MSIE")>0) {
        return "MSIE";                              //IE 浏览器
```





```
        }
        if(isFirefox=navigator.userAgent.indexOf("Firefox")>0){
            return "Firefox";                              //Firefox 浏览器
        }
        if(isSafari=navigator.userAgent.indexOf("Safari")>0) {
            return "Safari";                               //Safari 浏览器
        }
        if(isCamino=navigator.userAgent.indexOf("Camino")>0){
            return "Camino";                               //Camino 浏览器
        }
        if(isMozilla=navigator.userAgent.indexOf("Gecko/")>0){
            return "Gecko";                                //Gecko 浏览器
        }
    }
```

然后，在 Ajax 中调用自定义函数，通过自定义函数根据不同的浏览器类型执行不同格式的方法。其关键代码如下：

```
    var btype=getOs();
    xmlHttp.onreadystatechange = (btype!="Firefox")?(xmlHandle):(xmlHandle());
    xmlHttp.open ("GET",CartUrl,false);
    xmlHttp.send(null);
```

如此 Ajax 实现的功能在 IE6、IE7 或 Firefox 浏览器下就都可以正常运行了。

问题 220　jQuery 特效复选框的制作方法

问题阐述

表单中复选框的多选、反选是一个很常用的方法，经常出现在数据的后台管理中，实现多数据的批量操作。其最常用的实现方法就是通过 JavaScript 脚本来完成，编写相当长的一段 JavaScript 代码来实现这个功能，为了让这个功能实现得更加简洁、方便，下面介绍一种通过 jQuery 技术实现的复选框。

专家解答

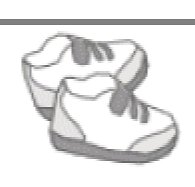
通过 jQuery 特性实现复选框的功能主要是应用 jQuery 中的事件，通过它们获取表单中的元素，根据元素名称进行判断，进而执行不同的操作。其主要应用的事件如下：

（1）$(document).ready()方法是事件模块中最重要的一个函数，它极大地提高了 Web 响应速度。$(document)用来获取整个文档对象，从这个方法的名称来理解，就是获取文档就绪的时候。该方法的书写格式为：

```
    $(document).ready(function() {
            //在这里写代码
    });
```

可以简写成：

```
    $().ready(function() {
            //在这里写代码
    });
```



当$()不带参数时，默认的参数就是 document，所以$()是$(document)的简写形式。

还可以进一步简写成：

```
$(function() {
        //在这里写代码
});
```

虽然语法可以更短一些，但是不提倡使用简写的方式，因为较长的代码更具可读性，同时也可以防止与其他方法混淆。

在 jQuery 中可以使用$(document).ready()方法代替传统的 window.onload()方法，不过两者之间还是存在一些细微的区别，主要表现在以下两个方面。

- ☑ 在一个页面上可以无限制地使用$(document).ready()方法，各个方法间并不冲突，会按照在代码中的顺序依次执行，而在一个页面中只能使用一个 window.onload()方法。
- ☑ 在一个文档完全下载到浏览器时（包括所有关联的文件，如图片、横幅等）就会响应 window.onload()方法。而$(document).ready()方法是在所有的 DOM 元素完全就绪以后就可以调用，不包括关联的文件。如在页面上还有图片没有加载完毕但是 DOM 元素已经完全就绪，这样就会执行$(document).ready()方法，在相同条件下 window.onload()方法是不会执行的，它会继续等待图片加载，直到图片及其他的关联文件都下载完毕时才执行。所以说$(document).ready()方法优于 window.onload()方法。

（2）在完成页面的加载之后，就需通过事件调用不同的方法实现不同的功能，这里应用的是 chick()事件，通过它调用一个函数，执行一个操作。

应用示例

通过 jQuery 特效实现复选框的功能，其关键代码如下：

```
<SCRIPT LANGUAGE="JavaScript">
$("document").ready(function(){
    $("#all").click(function(){
        if(this.checked){
            $("input[name='checkbox[]']").each(function(){this.checked=true;});
            $("#btn1").attr("value","反选");
        }else{
            $("input[name='checkbox[]']").each(function(){this.checked=false;});
            $("#btn1").attr("value","全选");
        }
    });
    $("#btn1").click(function(){
        $("[name='checkbox[]']").attr("checked",'true');          //全选
      })
    $("#btn2").click(function(){
        $("[name='checkbox[]']").removeAttr("checked");          //取消全选
    })
    $("#btn3").click(function(){
        $("[name='checkbox[]']:even").attr("checked",'true');    //选中所有奇数
    })
    $("#btn4").click(function(){
        $("[name='checkbox[]']").each(function(){
            if($(this).attr("checked")){
                $(this).removeAttr("checked");
```



```
                }else{
                    $(this).attr("checked",'true');
                }
            })
        })
    })
</SCRIPT>
```



专家点评

在 jQuery 中提供了很多的事件，通过它们可以完成很多操作，并且将在不同浏览器中的同一操作方法进行了统一，使其功能可以在不同的浏览器下正常运行。jQuery 中的事件如表 10.3 所示。

表 10.3　jQuery 中的事件

方　法	说　明
blur()	触发元素的blur事件
blur(fn)	在每一个匹配元素的blur事件中绑定一个处理函数，在元素失去焦点时触发，既可以是鼠标行为，也可以是使用Tab键离开的行为
change()	触发元素的change事件
change(fn)	在每一个匹配元素的change事件中绑定一个处理函数，在元素的值改变并失去焦点时触发
chick()	触发元素的chick事件
click(fn)	在每一个匹配元素的click事件中绑定一个处理函数，在元素上单击时触发
dblclick()	触发元素的dblclick事件
dblclick(fn)	在每一个匹配元素的dblclick事件中绑定一个处理函数，在某个元素上双击触发
error()	触发元素的error事件
error(fn)	在每一个匹配元素的error事件中绑定一个处理函数，当JavaSprict发生错误时，会触发error事件
focus()	触发元素的focus事件
focus(fn)	在每一个匹配元素的focus事件中绑定一个处理函数，当匹配的元素获得焦点时触发，通过鼠标单击或按Tab键触发
keydown()	触发元素的keydown事件
keydown(fn)	在每一个匹配元素的keydown事件中绑定一个处理函数，当键盘按下时触发
keyup()	触发元素的keyup事件
keyup(fn)	在每一个匹配元素的keyup事件中绑定一个处理函数，会在按键释放时触发
keypress()	触发元素的keypress事件
keypress(fn)	在每一个匹配元素的keypress事件中绑定一个处理函数，按下按键时触发（即按下并抬起同一个按键）
load(fn)	在每一个匹配元素的load事件中绑定一个处理函数，匹配的元素内容完全加载完毕后触发
mousedown(fn)	在每一个匹配元素的mousedown事件中绑定一个处理函数，鼠标在元素上单击后触发
mousemove(fn)	在每一个匹配元素的mousemove事件中绑定一个处理函数，鼠标在元素上移动时触发
mouseout(fn)	在每一个匹配元素的mouseout事件中绑定一个处理函数，鼠标从元素上离开时触发
mouseover(fn)	在每一个匹配元素的mouseover事件中绑定一个处理函数，鼠标移入对象时触发


Note


续表

方　法	说　明
mouseup(fn)	在每一个匹配元素的mouseup事件中绑定一个处理函数，鼠标单击对象释放时触发
resize(fn)	在每一个匹配元素的resize事件中绑定一个处理函数，当文档窗口改变大小时触发
scroll(fn)	在每一个匹配元素的scroll事件中绑定一个处理函数，当滚动条发生变化时触发
select()	触发元素的select事件
select(fn)	在每一个匹配元素的select事件中绑定一个处理函数，当用户在文本框（包括input和textarea）选中某段文本时触发
submit()	触发元素的submit事件
submit(fn)	在每一个匹配元素的submit事件中绑定一个处理函数，表单提交时触发
unload(fn)	在每一个匹配元素的unload事件中绑定一个处理函数，在元素卸载时触发

以上都是对应的 jQuery 事件，和传统的 JavaScript 中的事件几乎相同，只是名称不同。方法中的 fn 参数表示一个函数，事件处理程序就写在该函数中。

问题 221　如何使用 jQuery 技术制作级联菜单？

问题阐述

使用 Ajax 实现与服务器端进行交互，代码编写相对烦琐，为了简化 Ajax 操作，很多组织都推出了基于 Ajax 的框架，jQuery 就是其中代表之一。它是继 prototype 之后又一个优秀的 JavaScript 框架，其宗旨是写更少的代码，做更多的事情。那么，如何使用 jQuery 技术制作级联菜单呢？

专家解答

jQuery 实现 js 代码和 html 内容分离，不必在 html 中插入一堆 js 来调用命令，只需定义 id 即可。下面来了解 jQuery 中基本函数的应用以及其代表的意义。

（1）$()，用于替代 document.getElementById()。其代码如下：

```
var someElement = $("#myId");
```

通过 jQuery 获取 DOM 对象中的元素，示例代码如下：

```
$("div m");                    //
$("div.admin");                //
$("div #mysqlid ");            //
$("table a",content);          //
```

第 1 行代码获取所有标签下的 m 元素；第 2 行代码获取 class 为 admin 的元素；第 3 行代码获取标签下面 id 为 mysqlid 的元素；第 4 行代码获取 content 为上下文的 table 中所有的连接元素。

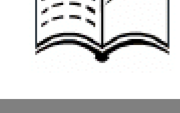
（2）ready(fn)，当 DOM 载入就绪可以查询及操纵时绑定一个要执行的函数。该方法是 window.load 事件的替代方法。

```
ready(fn)
```

参数 fn (Function)是在 DOM 就绪时执行的函数，可以为该参数任意起一个名字，并因此



可以不再担心命名冲突而放心地使用$别名。



注意：

在使用 ready(fn)时必须确保在<body>元素的 onload 事件中没有注册函数，否则将不会触发$(document).ready()事件。另外，可以在同一个页面中无限次地使用$(document).ready()事件。其中注册的函数会按照代码中的先后顺序依次执行。

通过$(document).ready(fn)代替 body 标签的 onload 事件，应用示例如下：

```
$(document).ready(function(){
      alert("jQuery");
});
```

使用$(document).ready()的简写，同时内部的 jQuery 代码依然使用$作为别名，而不必考虑全局的$是什么。其 jQuery 代码如下：

```
jQuery(function($) {
   //可以在这里继续使用$作为别名...
});
```

上述$(document).ready()的简写替换的是如下代码：

```
<body onload="alert('jQuery');">
```

（3）jQuery 中的事件机制。change()触发每个匹配元素的 change 事件。该函数会调用执行绑定到 change 事件的所有函数，包括浏览器的默认行为。可以通过在某个绑定的函数中返回 false 来防止触发浏览器的默认行为。change 事件会在元素失去焦点时触发，也会当其值在获得焦点后改变时触发。

change(fn)在每一个匹配元素的 change 事件中绑定一个处理函数。change 事件会在元素失去焦点时触发，也会当其值在获得焦点后改变时触发。

参数 fn (Function)是在每一个匹配元素的 change 事件中绑定的处理函数。

应用示例 1：为所有的文本框增加输入验证，代码如下：

```
$("input[type='text']").change( function() {
      //这里可以写些验证代码
});
```

click(fn)，在每一个匹配元素的 click 事件中绑定一个处理函数。参数 fn(Function)是绑定到 click 事件的函数。

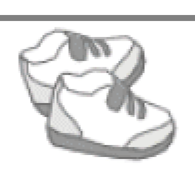
应用示例 2：将页面内所有段落点击后隐藏。其代码如下：

```
$(document).ready(function(){
      $("p").click( function () {
            $(this).hide();
      });
})
```

（4）jQuery 中的 Ajax 请求。通过远程 HTTP GET 请求载入信息。语法如下：

```
jQuery.get(url,[data],[callback])
```

这是一个简单的 GET 请求功能，以取代复杂$.ajax。请求成功时可调用回调函数。如果需





Note

要在出错时执行函数，则使用$.ajax。

GET 请求返回一个 XMLHttpRequest。参数 url(String)是待载入页面的 URL 地址；data(Map)为可选参数，是待发送 Key/value 参数；callback(Function)也是可选参数，是载入成功时的回调函数。

示例 1：请求 return.php 网页，忽略返回值。

```
$.get("return.php");
```

示例 2：请求 return.php 网页，传送两个参数，忽略返回值。

```
$.get("return.php", { name: "php", time: "2pm" } );
```

示例 3：显示 return.php 返回值（HTML 或 XML，取决于返回值）。

```
$.get("return.php", function(data){
  alert("Data Loaded: " + data);
});
```

示例 4：显示 return.php，传递两个参数，添加返回值。

```
$.get("return.php",{flag: "c",p: $("#p").val()}, function(data){
      $("#c").css("display","");
      $("#c").empty();
      $("#c").append(data);                    //将数据追加到城市下拉列表
      $("#cs").css("display","none");          //初始状态使城市下拉列表不可见
});
```

通过远程 HTTP POST 请求载入信息。

这是一个简单的 POST 请求功能，以取代复杂$.ajax。请求成功时可调用回调函数。如果需要在出错时执行函数，则使用$.ajax。

其返回值和参数的设置与 HTTP GET 相同，应用方法可以参考 GET 请求中的内容。

（5）jQuery 中的 CSS。css(name)，访问第一个匹配元素的样式属性。返回值为 string，参数 name 为要访问的属性名称。

示例 1：取得第一个段落的 color 样式属性的值。

```
$("p").css("color");
```

css(name,value)，在所有匹配的元素中设置一个样式属性的值。数字将自动转化为像素值。

参数 name 为属性名，参数 value (String, Number)是属性值。

示例 2：初始状态下隐藏下拉列表。

```
$("#cs").css("display","none");               //初始状态使城市下拉列表不可见
```

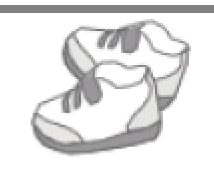
（6）jQuery 中的文档处理方法。empty()，删除匹配的元素集合中所有的子节点。其返回值是 jQuery。

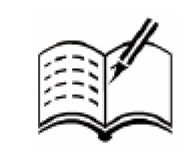
示例 3：清空下拉列表中的子元素。

```
$("#c").empty();
```

append(content)，向每个匹配的元素内部追加内容。该操作与对指定的元素执行 appendChild()方法，将它们添加到文档中的情况类似。

其返回值为 jQuery，参数 content (String, Element, jQuery)是要追加到目标中的内容。





示例 4：向所有段落中追加一些 HTML 标记。

```
<p>I would like to say: </p>
$("p").append("<b> Welcome to beijing</b>");
```

制作级联菜单的原理是：通过 jQuery 中的 get 发送请求；通过 append(content)向下拉列表中添加数据；通过 empty()清空下拉列表；通过 css(name,value)完成下拉列表的隐藏；通过 change(fn)事件实现下拉列表的关联操作。其关键代码如下：

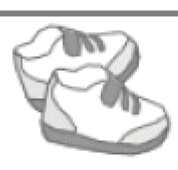
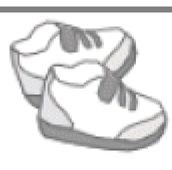
```
$(document).ready(
    function(){
        $.get("returnpc.php?flag=p", null, function(data){//向服务器发送 GET 请求，获取省份的值
            $("#p").append(data);                    //将结果追加到省份下拉列表中
        });
        $("#c").css("display","none");         //初始状态使城市下拉列表不可见
        $("#p").change(function(){             //为省份下拉列表增加改变事件
            if($("#p").val()==""){            //在没选择省份的情况下，使城市下拉列表不可见
                $("#c").css("display","none");
            }else{
                $.get("returnpc.php?flag=c&p="+$("#p").val(), null, function(data){
                                                //如果选择了某省份发送 GET 请求
                    $("#c").css("display","");      //并使城市下拉列表可见
                $("#c").empty();
                    $("#c").append(data);           //将数据追加到城市下拉列表
                $("#cs").css("display","none");     //初始状态使城市下拉列表不可见
                });
            }
        });
        $("#cs").css("display","none");                 //初始状态使城市下拉列表不可见
        $("#c").change(function(){
            if($("#c").val()==""){
                $("#cs").css("display","none");
            }else{
$.get("returnpc.php?flag=cs&c="+$("#c").val()+"&p="+$("#p").val(),null,function(data){
                $("#cs").css("display","");
                $("#cs").empty();
                    $("#cs").append(data);          //将数据追加到城市下拉列表
                });
            }
        });
    }
);
```

应用示例

本示例中应用 jQuery 制作一个级联菜单，实现省、市和县的 3 级联动，其具体步骤如下：

（1）创建 index.php 文件，完成主页面的设计。首先通过 script 标签调用 jQuery 框架包文件 jquery-1.3.2.js 以及 fun.js 文件。然后，通过<select></select>标签，创建 3 个下拉列表，用于输出省、市和县的内容，分别定义它们的 id 为 p、c 和 cs。其关键代码如下：

```
<script language="javascript" src="js/jquery-1.3.2.js"></script>
<script language="javascript" src="js/fun.js"></script>
```


```
        <tr>
            <td height="50" colspan="2" align="center" valign="middle">查询全国各个城市的天气情况：
                <select id="p" name="p" >
                    <option value="">-请选择地区-</option>
                </select>
                <select id="c" name="c"> </select>
                <select id="cs" name="cs"> </select>
            </td>
        </tr>
```

（2）创建 jQuery 中 GET 请求的 returnpc.php 文件，在 GET 请求中传递的是省、市和县的 id。首先，通过 require_once 调用存储省、市和县数据的文件 pc.php，获取 GET 传递的参数，定义变量 str。代码如下：

```
<?php
require_once 'pc.php';                          //包含省份数组和城市数组
$flag = $_GET['flag'];                          //用于区分是返回省份信息还是返回城市信息
$str="";
```

然后，判断当参数值为 p 时，将数组 p 中存储的省份信息添加到省份下拉列表中。将数据赋给变量 str，并输出变量 str。代码如下：

```
if($flag == p){                                 //如果参数 flag 的值为 p，则返回省份信息
    for($i=0; $i<count($p); $i++){              //通过循环整理所有要追加到省份下拉列表的内容
        $str.="<option
value=\"".iconv('gbk','utf-8',$p[$i])."\">".iconv('gbk','utf-8',$p[$i])."</option>";
    }
    echo $str;                                  //输出要返回的参数
```

接着，判断当参数值为 c 时，表示返回的是市的信息，提取参数值 p 所对应的省份在省份数组中的键值，根据省份的键值确定所对应的市，并通过 for 循环将市数组 c 中的数据添加到市下拉列表中。将数据赋给变量 str，并输出变量 str。代码如下：

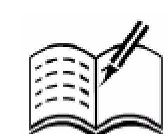
```
}elseif ($flag==c){                             //如果 flag 的值等于 c，表示返回的是市级城市
        $index = array_search($_GET['p'], $p); //提取所选择省份的在省份数组中对应的键值
    for($j=0; $j<count($c[$index]); $j++){      //通过循环整理所有要追加到城市下拉列表的内容
        $str.= "<option value=\"".iconv('gbk','utf-8',$c[$index][$j])."\">".iconv('gbk','utf-8',$c[$index][$j]).
"</option>";
    }
    echo $str;                                  //输出要返回的参数
```

最后，判断当参数值为 cs 时，表示返回的是县的信息，分别提取参数值 p 所对应的省份在省份数组中的键值和参数值 c 所对应的市在市数组中的键值，根据省份的键值和市的键值确定所对应的县，最终通过 for 循环将县数组 cs 中的数据添加到县下拉列表中。将数据赋给变量 str，并输出变量 str。代码如下：

```
}else{
    $index = array_search($_GET['p'], $p);              //提取所选择省份的在省份数组中对应的键值
    $indexs = array_search($_GET['c'], $c[$index]);//提取所选择省份的在市城市数组中对应的键值
    for($k=0; $k<count($cs[$index][$indexs]); $k++){    //通过循环整理所有要追加到城市下拉
                                                          列表的内容
        $str.=  "<option  value=\"".iconv('gbk','utf-8',$cs[$index][$indexs][$k])."\">".iconv('gbk','utf-8',
```



```
$cs[$index][$indexs][$k])."</option>";
                }
                echo $str;                                              //输出要返回的参数
        }
        ?>
```



（3）创建 pc.php 文件，在该文件中包含 p、c 和 cs 3 个数组，分别用于存储省、市和县的数据。运行本示例，运行结果如图 10.9 所示。

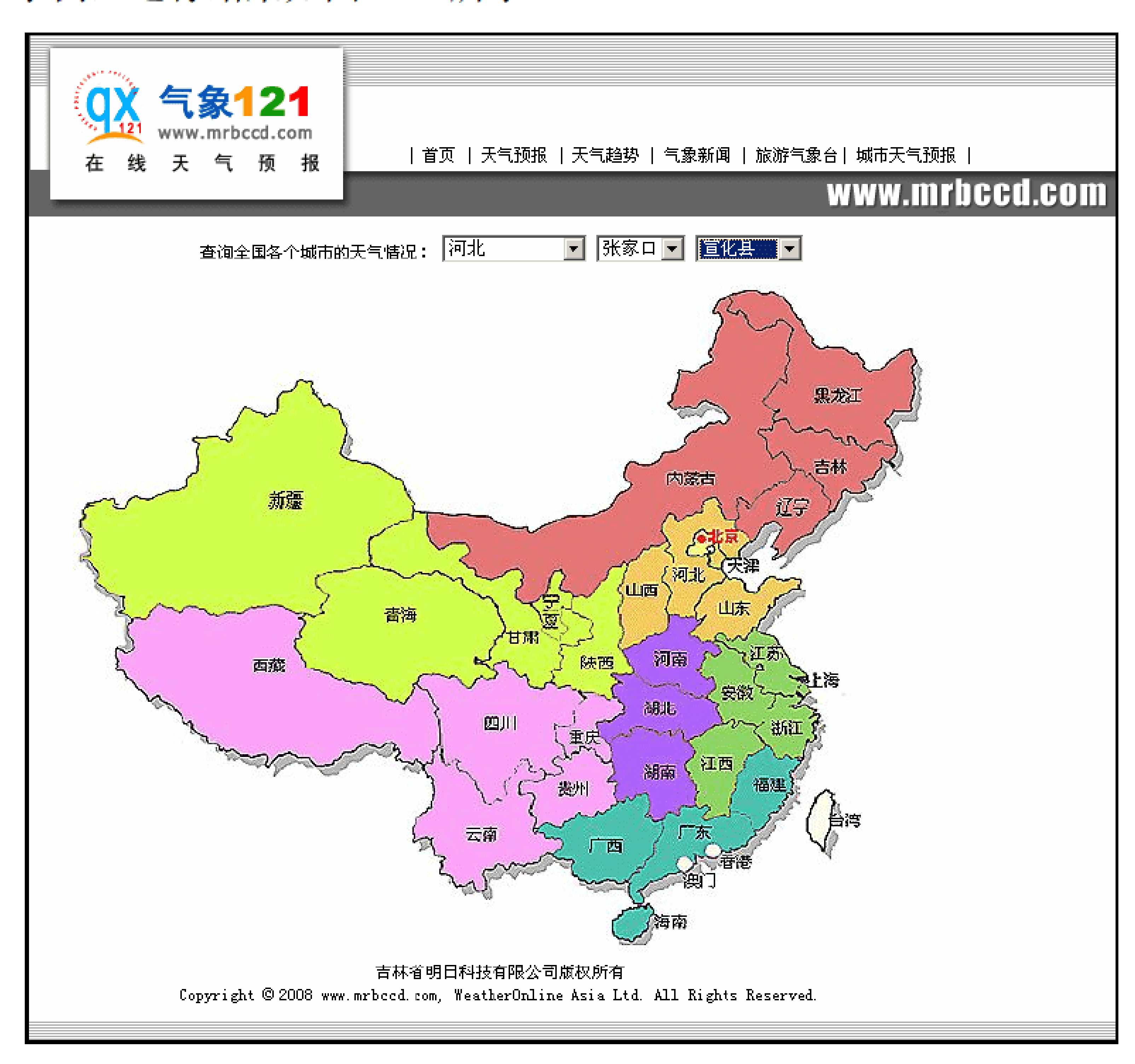


图 10.9　jQuery 制作级联菜单

专家点评

要应用 jQuery 技术，其前提是必须在程序中包含 jQuery 的框架包。本示例中使用的是 jquery-1.3.2.js。

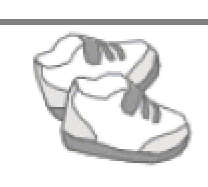
问题 222　如何通过 jQuery 生成网页特效？

问题阐述

在 JavaScript 中，生成网页特效编写的代码往往非常繁琐、复杂，而通过 jQuery 来生成网页的特效却非常简单，而且代码也非常少。那么，它具体的操作步骤是怎样的呢？又应用到哪些技术呢？

专家解答

通过 jQuery 生成网页特效主要应用的是 jQuery 中的特效函数，下面对 jQuery 中的常用特





效函数进行讲解。

（1）基本效果，实现匹配元素的隐藏和显示。hide()隐藏显示的元素。它是 hide(speed, [callback])的无动画版。如果选择的元素是隐藏的，则该方法将不会改变任何东西。

示例：隐藏所有段落。

```
$("p").hide()
```

hide(speed,[callback])以优雅的动画隐藏所有匹配的元素，并在显示完成后有选择地触发一个回调函数，同时可以根据指定的速度动态地改变每个匹配元素的高度、宽度和不透明度。

参数 speed(String,Number)表示 3 种预定速度之一的字符串("slow", "normal" or "fast")或者表示动画时长的毫秒数值（如 1000）。

参数 callback(FunctionFunction) : (Optional)表示在动画完成时执行的函数，每个元素执行一次。

示例 1：用 600 毫秒的时间将段落缓慢隐藏。

```
$("p").hide("slow");
```

示例 2：用 200 毫秒将段落迅速隐藏，并弹出一个对话框。

```
$("p").hide("fast",function(){
    alert("Welcome To BEI JING.");
 });
```

show()显示隐藏的匹配元素。

它是 show(speed, [callback])无动画的版本。如果选择的元素是可见的，则该方法将不会改变任何东西。无论这个元素是通过 hide()方法隐藏的，还是在 CSS 中设置了 display:none;，该方法都将有效。

示例：显示所有段落。

```
<p style="display: none">PHP</p>
$("p").show()
```

show(speed,[callback])以优雅的动画显示所有匹配的元素，并在显示完成后有选择地触发一个回调函数，同时可以根据指定的速度动态地改变每个匹配元素的高度、宽度和不透明度。

参数 speed (String,Number)表示 3 种预定速度之一的字符串("slow", "normal" or "fast")或者表示动画时长的毫秒数值（如 1000）。

参数 callback (Function) : (Optional)表示在动画完成时执行的函数，每个元素执行一次。

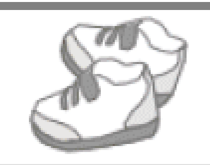
示例 1：用缓慢的动画将隐藏的段落显示出来，用时 600 毫秒。代码如下：

```
<p style="display: none">PHP</p>
$("p").show("slow");
```

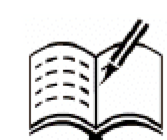
示例 2：迅速将隐藏的段落显示出来，用时 200 毫秒。执行后调用回调函数。代码如下：

```
<p style="display: none">Hello</p>
$("p").show("fast",function(){
    $(this).text("Welecome To BEIJING!");
 });
```

示例 3：将隐藏的段落用将近 4 秒的时间显示出来，执行后调用回调函数。代码如下：



```
<p style="display: none">Hello</p>
$("p").show(4000,function(){
    $(this).text("Welcome To BeiJing...");
 });
```



toggle()切换元素的可见状态。如果元素是可见的，则切换为隐藏的；如果元素是隐藏的，则切换为可见的。

示例：切换所有段落的可见状态。代码如下：

```
<p>PHP</p><p style="display: none">Welcome To BeiJing! </p>
$("p").toggle()
```

输出结果：<p tyle="display: none">PHP</p><p style="display: block"> Welcome To BeiJing! </p>。

（2）淡入、淡出效果。fadeIn(speed,[callback])通过不透明度的变化来实现所有匹配元素的淡入效果，并在动画完成后有选择地触发一个回调函数。

这个动画只调整元素的不透明度，也就是说，所有匹配的元素的高度和宽度不会发生变化。

参数 speed (String,Number)表示 3 种预定速度之一的字符串("slow", "normal" or "fast")或者表示动画时长的毫秒数值（如 1000）。

参数 callback (Function) : (Optional)为可选参数，表示在动画完成时执行的函数。

示例 1：用 600 毫秒缓慢将段落淡入。代码如下：

```
$("p").fadeIn("slow");
```

示例 2：用 200 毫秒快速将段落淡入，并弹出一个对话框。代码如下：

```
$("p").fadeIn("fast",function(){
    alert("Welcome To BeiJing.");
 });
```

参数 fadeOut(speed,[callback])通过不透明度的变化来实现所有匹配元素的淡出效果，并在动画完成后有选择地触发一个回调函数。

这个动画只调整元素的不透明度，也就是说，所有匹配的元素的高度和宽度不会发生变化。

参数 speed (String,Number)表示 3 种预定速度之一的字符串("slow", "normal" or "fast")或者表示动画时长的毫秒数值（如 1000）。

参数 callback (Function)为可选参数，表示在动画完成时执行的函数

示例 1：用 600 毫秒缓慢地将段落淡出。代码如下：

```
$("p").fadeOut("slow");
```

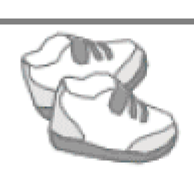
示例 2：用 200 毫秒迅速地将段落淡出，并弹出一个对话框。代码如下：

```
$("p").fadeOut("fast",function(){
    alert("Welcome To BeiJing.");
 });
```

fadeTo(speed,opacity,[callback])把所有匹配元素的不透明度以渐进方式调整到指定的不透明度，并在动画完成后有选择地触发一个回调函数。

这个动画只调整元素的不透明度，也就是说，所有匹配元素的高度和宽度不会发生变化。

参数 speed (String,Number)表示 3 种预定速度之一的字符串("slow", "normal" or "fast")或者





Note

表示动画时长的毫秒数值（如 1000）。

参数 opacity (Number)表示要调整到的不透明度值（0~1 之间的数字）。

参数 callback (Function)为可选参数，为在动画完成时执行的函数。

示例：用 600 毫秒缓慢地将段落的透明度调整到 0.66，大约 2/3 的可见度。代码如下：

```
$("p").fadeTo("slow", 0.66);$("p").fadeTo("slow", 0.66);
```

（3）滑动效果。slideDown(speed,[callback])通过高度变化（向下增大）来动态地显示所有匹配的元素，在显示完成后可以有选择地触发一个回调函数。

这个动画效果只调整元素的高度，可以使匹配的元素以“滑动”的方式显示出来。

参数 speed (String,Number)表示 3 种预定速度之一的字符串("slow", "normal" or "fast")或者表示动画时长的毫秒数值（如 1000）。

参数 callback (Function)为可选参数，为在动画完成时执行的函数。

示例 1：用 600 毫秒缓慢的将段落滑下。代码如下：

```
$("p").slideDown("slow");
```

示例 2：用 200 毫秒迅速地将段落滑下，并弹出一个对话框。代码如下：

```
$("p").slideDown("fast",function(){
    alert("Welcome To BeiJing.");
 });
```

slideToggle(speed,[callback])通过高度变化来切换所有匹配元素的可见性，并在切换完成后有选择地触发一个回调函数。

这个动画效果只调整元素的高度，可以使匹配的元素以“滑动”的方式隐藏或显示。

参数 speed (String,Number)表示 3 种预定速度之一的字符串("slow", "normal" or "fast")或表示动画时长的毫秒数值（如 1000）。

参数 callback (Function)为可选参数，表示在动画完成时执行的函数。

示例 1：用 600 毫秒缓慢地将段落滑上或滑下。代码如下：

```
$("p").slideToggle("slow");
```

示例 2：用 200 毫秒迅速地将段落滑上或滑下，并弹出一个对话框。代码如下：

```
$("p").slideToggle("fast",function(){
    alert("welecome To BeiJing.");
 });
```

slideUp(speed,[callback])通过高度变化（向上减小）来动态地隐藏所有匹配的元素，在隐藏完成后有选择地触发一个回调函数。

这个动画效果只调整元素的高度，可以使匹配的元素以“滑动”的方式隐藏起来。

参数 speed (String,Number)表示 3 种预定速度之一的字符串("slow", "normal" or "fast")或表示动画时长的毫秒数值（如 1000）。

参数 callback (Function)为可选，表示在动画完成时执行的函数。

示例 1：用 600 毫秒缓慢地将段落滑上。代码如下：

```
$("p").slideUp("slow");
```

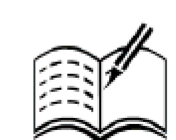
示例 2：用 200 毫秒迅速地将段落滑上，并弹出一个对话框。代码如下：



```
        $("p").slideUp("fast",function(){
            alert("Welecome To BeiJing.");
         });
```

应用示例

本示例应用 jQuery 实现网页中图片的淡入和淡出的效果，并且实现数据的间断性滑动更换。其具体实现步骤如下：



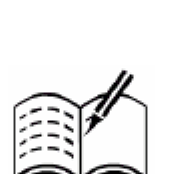
（1）创建 js 文件夹，载入 jQuery 框架包；创建 js 脚本文件，编写自定义函数，通过 jQuery 特效函数定义网页特效功能。在本示例中实现 3 种效果，图片的淡入和淡出以及页面内容的滑动更换。jQuery.js 脚本文件的关键代码如下：

```
function A(){                                          //创建自定义函数
      $("#a").fadeOut("slow",function(){               //实现图片的淡出效果
               $("#e").fadeIn("slow");                 //执行回调函数，实现另外一个图片的淡入
        });
      $("#aa").fadeOut("slow");                        //执行图片的淡出操作
      $("#cc").fadeOut("slow");
      $("#dd").fadeOut("slow");
      $("#bb").fadeIn("slow");
}
setTimeout("A()", 10800);                              //间隔一段时间执行一次函数
function B(){
      $("#b").fadeOut("slow",function(){
               $("#f").fadeIn("slow");
        });
      $("#aa").fadeOut("slow");
      $("#bb").fadeOut("slow");
      $("#dd").fadeOut("slow");
      $("#cc").fadeIn("slow");
}
```

（2）创建 index.php 页面，通过 Script 脚本包含 js 脚本文件和 jQuery 框架包，应用 onClick 事件调用自定义函数，实现图片的淡入和淡出以及内容的替换。其关键代码如下：

```
<script src="js/jquery-1.3.2.js"></script>
<script src="js/jquery.js"></script>
<tr>
          <td width="450" align="right">
      <img src="images/dl_08-(9)_01.jpg" width="11" height="164" />
      <img src="images/dl_08-(9)_02.jpg" name="a" width="106" height="164" border="0" usemap="#aMap" id="a" style="display:inline" onclick="A();" />
      <img src="images/dl_08-(9)_03.jpg" width="109" height="164" border="0" usemap="#bMap" id="b" style="display:inline" onclick="B();" />
……省略部分代码
      <img id="h" src="images/dl_08-(9)_05.jpg" width="113" height="164" style="display:none" onclick="H();" border="0" /></td>
      <td width="350"><img src="images/dl_08(10).jpg" width="350" height="164" /></td>
</tr>
<tr>
      <td width="450" height="180" align="right" valign="top" id="aa" style="display:line" >
          <table width="448" border="0" cellpadding="0" cellspacing="0">
```





Note

```
                    <tr>
                        <td width="11" height="30"> </td>
                        <td width="427" align="center"><img src="images/dl_082.jpg" width="401"
height="28" /></td>
                        <td width="10"> </td>
                    </tr>
                    <tr>
                        <td> </td>
                        <td height="150" valign="top">
                        <table width="427" border="0" cellpadding="0" cellspacing="0">
                            <tr>
                                <td width="45" height="10"> </td>
                                <td width="167"> </td>
                                <td width="215"> </td>
                            </tr>
                <?php
                    include_once("conn/conn.php");
                    $result=mysql_query("select * from tb_articles",$conn);
                  while($myrow=mysql_fetch_array($result)){
                ?>
                            <tr>
                                <td height="22" align="right" class="STYLE3">.</td>
                                <td align="left" class="STYLE3"><a href="#"><?php echo
$myrow[topic];?></a></td>
                                <td class="STYLE3"><?php echo $myrow[addtime];?></td>
                            </tr>
              <?php
                  }
              ?>
                          </table></td>
                        <td> </td>
                    </tr>
                  </table></td>
         </tr>
```

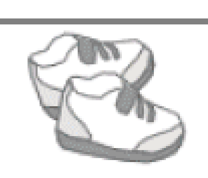
运行结果如图 10.10 所示。

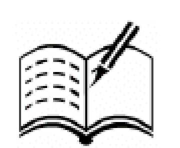


图 10.10　jQuery 生成网页特效



XML 技术





Note

问题 223　如何处理 XML 中的数据刷新的问题？

问题阐述

在调用 XML 文件中的数据时，由于数据经常更新，所以必须刷新 XML 文件才能输出最新的数据。但是在刷新 XML 文件的过程中，直接刷新页面不能对 XML 文件中的数据进行更新，而必须要关闭网页，再重新打开，才能完成 XML 文件中数据的刷新，操作起来非常烦琐。那么，如何解决 XML 中的数据刷新的问题呢？

专家解答

解决此问题的方法为：

（1）在浏览器中选择“工具”→“Internet 选项”命令。

（2）选择“常规”选项卡，在该页面中单击“设置”按钮，打开“设置”窗口。

（3）在“设置”窗口中选中“每次访问此页时检查”单选按钮，如图 11.1 所示。

（4）单击“确定”按钮，完成设置。



图 11.1　“设置”窗口

问题 224　RSS 2.0 都包括哪些元素？

问题阐述

什么是 RSS？RSS 2.0 都包括哪些元素？

专家解答

RSS 为 Really Simple Syndication（简易供稿）的缩写，是某一站点用来和其他站点之间共享内容的一种简易方式，也叫聚合内容。网络用户可以在客户端借助于支持 RSS 的新闻聚合工具软件（如 SharpReader NewzCrawler、FeedDemon RSSReader），在不打开网站内容页面的情况下阅读支持 RSS 输出的网站内容。

从 RSS 阅读者的角度来看，完全不必考虑它到底是什么意思，只要简单地理解为一种方便的信息获取工具就可以了。RSS 获取信息的模式与加入邮件列表（如电子杂志和新闻邮件）获取信息有一定的相似之处，也就是可以不必登录各个提供信息的网站而通过客户端浏览方式（称为“RSS 阅读器”）或者在线 RSS 阅读方式浏览内容。

1. RSS 的历史

RSS 最早的版本是 0.90，是由 Netscape 公司设计的，目的是用来建立一个整合了各主要新闻站点内容的门户，但是 0.90 版本的 RSS 规范过于复杂，随后简化的 RSS 0.91 版本也随着 Netscape 公司对该项目的放弃而于 2000 年暂停。

后来，一家专门从事博客写作软件开发的公司 UserLand 接手了 RSS 0.91 版本的发展，并把它作为其博客写作软件的基础功能之一继续开发，逐步推出了 0.92、0.93 和 0.94 版本。随着



网络博客的流行，RSS 作为一种基本的功能也被越来越多的网站和博客软件支持。

在 UserLand 公司接手并不断开发 RSS 的同时，很多的专业人士意识到需要通过一个第三方、非商业的组织，把 RSS 发展成为一个通用的规范，并进一步标准化。于是 2001 年一个联合小组在 0.90 版本 RSS 的开发原则下，以 W3C 新一代的语义网技术 RDF（Resource Description Framework）为基础，对 RSS 进行了重新定义，发布了 RSS 1.0，并将 RSS 定义为“RDF Site Summary”。但是这项工作没有与 UserLand 公司进行有效的沟通，UserLand 公司也不承认 RSS 1.0 的有效性，并坚持按照自己的设想进一步开发出 RSS 的后续版本，到 2002 年 9 月发布了最新版本 RSS 2.0，UserLand 公司将 RSS 定义为“Really Simple Syndication”。

到目前为止，RSS 从 0.91 起到 2.01，一共有 7 个版本之多。各个版本除 1.0 以外大同小异。

在 RSS 阅读器中读取的是站点中一个标准格式的 RSS 2.0 文件，所以说要让站点支持 RSS 阅读，标准 RSS 2.0 格式文件的创建是关键。RSS 2.0 结构可以划分为 3 个部分。

- ☑ RSS 的根元素 rss。
- ☑ 在根元素下的 channel 元素（表示新闻频道的特性）。
- ☑ 在根元素下 channel 元素之后出现的 item 元素（表示新闻条目）。

2. RSS 的根元素

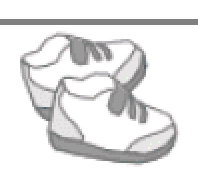
rss 元素是 RSS 的根元素，具有 version 属性，可以包含任何名称空间的其他属性。在该元素下的子元素有两个：channel（频道，只能出现一次）和 item（条目，可以出现多次）。

3. channel（频道）元素及其子元素

channel 元素表示网站上所发布的新闻栏目，该元素的子元素如表 11.1 所示。

表 11.1　channel 元素的可用子元素

元素名称	说　明
title	网站上新闻频道的标题，该标题应该与html网站的标题保持一致
link	可以到达此频道相关的HTML网页的超链接
description	频道的描述
language	频道对应的语种
copyright	网站著作权的声明
managingEditor	频道内容编辑的电子邮件地址
webMaster	维护网站频道的主要技术人员
pubDate	发布RSS的时间
lastBuilDate	最后更新网站内容的时间
category	该栏目所属的类别，此元素包含“domain”属性
generator	生成RSS栏目的程序
docs	与RSS所执行的版本标准对应的URL（很少用）
cloud	表示应用程序注册订阅此频道的信息，RSS应用程序使用此元素的信息，自动检查RSS更新。此元素的子元素如下： domain：自动下载RSS时的域名 port：下载RSS时所采用的端口 path：RSS在域名下的路径 registerProcedure：订阅RSS时所执行的程序 protocol：传输下载RSS时所用的网络协议，如HTTP等







Note

续表

元素名称	说　明
ttl	RSS内容在客户浏览器中缓存的时间
image	与频道对应的图标
textInput	可以用来接受文本的输入框，子元素如下： title：提交按钮上的文字 description：对此文本框的描述 name：文本框的名称 link：服务器上处理文本请求的应用程序路径
skipHours	RSS下载程序不检查RSS更新的时间，此元素包含hour子元素（最多24个），表示不检查RSS更新的钟点，hour元素值所代表的每个时间对应格林威治时间（GMT，Greenwich Mean Time）的小时
skipDays	RSS下载程序可以跳跃不检查RSS更新的时期，此元素包含day子元素（最多7个），表示不检查RSS更新的日期，day元素的值分别用Monday、Tuesday、Wednesday、Thursday、Friday、Saturday和Sunday表示星期一到星期日

4. item（项目）元素及其子元素

item 元素表示网站上所发布的新闻，该元素的子元素如表 11.2 所示。

表 11.2　item 元素的可用子元素

元素名称	说　明
title	新闻标题
link	阅读新闻正文的超链接
description	新闻内容的摘要
author	本条目作者的电子邮件地址
category	此新闻所属的类别，包含domain属性（关于此类别的领域名称）
comments	此新闻条目对应的评论网页的超链接
enclosure	与此条目相关的多媒体的内容，此元素包含url（下载附件的超文本链接路径）、length（附件字节大小）和type（附件的MIME媒体类型）属性
guid	全球唯一标识符（Globally Unique Identifier）或永久的URL值，此元素包含isPermaLink（是否为永久的URL值）属性
pubDate	发布此新闻的时间
source	此新闻的来源，此元素包含url（此新闻所引用的频道的路径）属性

专家点评

上述介绍的内容就是 RSS 2.0 中包含的元素，其中，在生成 RSS 文件时不会应用到所有的元素，而是根据实际的需求应用不同的元素。

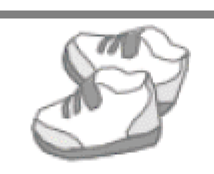
在 RSS 阅读器中读取的 RSS 文件中包含的元素如下：

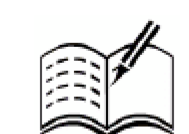
☑ rss 根元素。

☑ channel（频道）元素。该元素包含如下 3 个子元素。

title：网站新闻频道的标题。

link：此频道相关的 HTML 超链接。





description：频道内容的摘要。

☑ item（项目）元素。该元素包含如下 4 个子元素。

title：新闻标签。

link：阅读新闻正文的超链接。

description：新闻内容的摘要。

pubDate：新闻的发布日期。

问题 225　如何创建支持 RSS 阅读的站点？

问题阐述

因为 PHP 是面向 Web 开发的，而 RSS 阅读器这个模块似乎更适合通过应用程序来完成，所以作为 PHP 程序员更应该了解的是如何让网站支持 RSS 阅读，而不是如何更好地去创建开发 RSS 阅读器。还有一个更直接的原因就是，如果想使用 RSS 阅读器，似乎从网上下载更方便，更快捷。

专家解答

要让站点支持 RSS 阅读，必须考虑以下几个问题。

（1）将什么样的数据作为 RSS 阅读的内容。

（2）RSS 阅读器读取的是什么格式的文件。

（3）如何将数据库中的数据写入文件中。

（4）如何才能保证 RSS 阅读器中读取的是最新的数据。

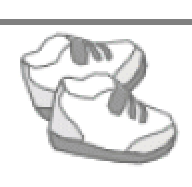
创建支持 RSS 阅读的站点

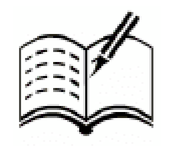
带着上述的疑问开始创建一个能够支持 RSS 阅读的程序，这里创建的是一个用于统计编程技巧的程序。通过该程序将每个人在编程中总结出的经验技巧进行统计，并存储到数据库中，然后再输出这些技巧让大家共享资源。其运行结果如图 11.2 所示。



图 11.2　编程技巧整合的运行结果

有关该程序的创建操作，这里将不作讲解，完整代码可参考配套资源中的内容。这里将编程技巧的标题和内容作为 RSS 阅读的内容，所以接下来要做的就是如何将编程技巧的标题和内





容添加到 RSS 文件中，并且保证 RSS 文件中的数据是最新的。

首先，保证 RSS 文件中的内容是最新的，方法为在数据被添加后直接更新 RSS 文件。

在 index_ok.php 中执行向数据库中添加编程技巧的操作，并调用 rss.php 文件重新生成一个标准格式的 XML 文件，同时更新 XML 文件中的数据。index _ok.php 的关键代码如下：

```
<?php
header("Content-Type:text/html; charset=utf-8");        //设置编码格式
include_once("conn/conn.php");                           //包含数据库连接文件
$dates=date("Y-m-d");                                    //设置时间
if($_POST['submit']=="提交"){                            //判断是否执行提交操作
    $query="insert into tb_rss_database (tb_rss_subject,tb_rss_content,tb_rss_author,tb_rss_department,
tb_rss_date) values('".$_POST['zhuti']."','".$_POST['neirong']."','".$_POST['user']."','".$_POST['bumen']."','$dates')";
                                                         //定义添加语句
    $result=mysql_query($query,$conn);                   //执行添加操作
    if($result){
        include_once("rss.php");                         //包含生成 RSS 的文件，重新生成 RSS 文件
        echo "<script> alert ('文件上传成功!'); window.location.href='index.php';</script>";
    }else{
        echo "<script> alert ('文件上传失败!'); window.location.href='index.php';</script>";
    }
}
?>
```

然后，将数据添加到 RSS 文件中，并且生成一个标准格式的 RSS 2.0 文件，文件的后缀是.xml。创建 XML 文件的操作是在 rss.php 文件中进行的，其操作步骤如下：

（1）通过服务器变量获取头信息的内容和当前页面前一页的 URL 地址以及当前的时间。代码如下：

```
$server_ip=$_SERVER['SERVER_NAME'];                     //获取服务器 IP 地址
$request_url=str_ireplace("index_ok.php","look_content.php",$_SERVER['REQUEST_URI']);
                                                        //获取实例的相对路径
$http="http://".$server_ip.$request_url;                //定义一个访问路径
$date_time=date("Y-m-d H:i:s");
```

（2）通过 new DomDocument 创建 DOM 对象，创建 XML 文件的根节点，并且为根节点添加属性及属性值。代码如下：

```
$dom = new DomDocument('1.0','utf-8');                  //创建 DOM 对象
$object = $dom->createElement('rss');                   //创建根节点 rss
$dom->appendChild($object);                             //将创建的根节点添加到 DOM 对象中
    $type1 = $dom->createAttribute('xmlns:rdf');        //创建一个节点属性 xmlns:rdf
    $object->appendChild($type1);                       //将属性追加到 rss 根节点中
        $type1_value = $dom->createTextNode('http://www.w3.org/1999/02/22-rdf-syntax-ns#');
                                                        //创建一个属性值
        $type1->appendChild($type1_value);              //将属性值赋给属性 xmlns:rdf
    $type2 = $dom->createAttribute('xmlns:dc');         //创建一个节点属性 xmlns:dc
    $object->appendChild($type2);                       //将属性追加到 rss 根节点中
        $type2_value = $dom->createTextNode('http://purl.org/dc/elements/1.1/');
                                                        //创建一个属性值
        $type2->appendChild($type2_value);              //将属性值赋给属性 xmlns:dc
    $type3 = $dom->createAttribute('xmlns:taxo');       //创建一个节点属性 xmlns:taxo
```





```
        $object->appendChild($type3);                    //将属性追加到 rss 根节点中
            $type3_value = $dom->createTextNode('http://purl.org/rss/1.0/modules/taxonomy/');
                                                         //创建一个属性值
            $type3->appendChild($type3_value);           //将属性值赋给属性 xmlns:taxo
        $type4 = $dom->createAttribute('version');       //创建一个节点属性 version
        $object->appendChild($type4);                    //将属性追加到 rss 根节点中
            $type4_value = $dom->createTextNode('2.0');  //创建一个属性值
            $type4->appendChild($type4_value);           //将属性值赋给属性 version
```

（3）创建 channel 节点，这是 RSS 2.0 中的 channel 元素及其子元素。其中包括 title、link、description 和 dc:creator 4 个子元素。代码如下：

```
    $channel = $dom->createElement('channel');                  //创建节点 channel
        $object->appendChild($channel);                         //将节点 channel 追加到 rss 根节点下
            $title = $dom->createElement('title');              //创建节点 title
            $channel->appendChild($title);                      //将节点追加到 channel 节点下
                $title_value = $dom->createTextNode('明日科技');//创建元素值
                $title->appendChild($title_value);                   //将值赋给 title 节点
            $link = $dom->createElement('link');                     //创建节点 link
            $channel->appendChild($link);                            //将节点追加到 channel 节点下
                $link_value = $dom->createTextNode('http://www.mingrisoft.com');     //创建元素值
                $link->appendChild($link_value);                     //将值赋给 link 节点
            $description = $dom->createElement('description');       //创建节点 description
            $channel->appendChild($description);                     //将节点追加到 channel 节点下
                $description_value = $dom->createTextNode('明日科技');     //创建元素值
                $description->appendChild($description_value);   //将值赋给 description 节点
            $dc_creator = $dom->createElement('dc:creator');         //创建节点 dc:creator
            $channel->appendChild($dc_creator);                      //将节点追加到 channel 节点中
                $dc_creator_value = $dom->createTextNode('http://www.mingrisoft.com');//创建元素值
                $dc_creator->appendChild($dc_creator_value);     //将值赋给 dc:creator 节点
```

（4）创建 item 节点。从数据库中读取出编程技巧的数据，并且应用 while 循环输出数据库中的数据，将其作为 item 节点中子节点的元素值。这里创建的是 RSS 2.0 中的 item 元素及其子元素，其中包括 title、link、description 和 pubDate 4 个子元素。代码如下：

```
    $sql=mysql_query("select * from tb_rss_database order by tb_rss_id desc");  //从数据库中读取数据
    while($myrow=mysql_fetch_array($sql)){                                       //循环输出数据库中数据
            $item = $dom->createElement('item');                     //创建节点 item
            $channel->appendChild($item);                            //将 item 追加到 channel 节点下
                $item_title = $dom->createElement('title');          //创建 title 节点
                $item->appendChild($item_title);                     //将节点追加到 item 节点下
                $item_link = $dom->createElement('link');            //创建 link 节点
                $item->appendChild($item_link);                      //将节点追加到 item 节点下
                $item_description = $dom->createElement('description');//创建 description 节点
                $item->appendChild($item_description);                   //将节点追加到 item 节点中
                $item_pubDate = $dom->createElement('pubDate');          //创建节点 pubDate
                $item->appendChild($item_pubDate);                   //将节点追加到 item 节点下
                    $title_value = $dom->createTextNode($myrow['tb_rss_subject']);     //创建元素值
                    $item_title->appendChild($title_value);          //将值赋给 title 节点
                    $link_value = $dom->createTextNode($http."?lmbs=".$myrow['tb_rss_id']);
                                                                     //创建元素值
                    $item_link->appendChild($link_value);            //将值赋给 link 节点
```


Note

```
                    $description=mb_substr($myrow['tb_rss_content'],0,80,"utf-8");
                                                                    //截取该字段中的前 80 个字符
                    $description_value = $dom->createTextNode($description);          //创建元素值
                    $item_description->appendChild($description_value); //将值赋给 description 节点
                    $pubDate_value = $dom->createTextNode($date_time);                //创建元素值
                    $item_pubDate->appendChild($pubDate_value);         //将值赋给 pubDate 节点
    }
```

（5）通过$dom->saveXML()方法生成 XML 文档，并应用 file_put_contents()函数将生成的 XML 文档保存到 Rss.xml 文件中。代码如下：

```
$modi = $dom->saveXML();                             //生成 XML 文档
file_put_contents('Rss.xml',$modi);                  /*将对象保存到 Rss.xml 文档中*/
```

至此，一个支持 RSS 阅读的站点创建完毕，只要在站点中添加新的数据，存储在根目录下的 XML 文件就会被更新一次，从而保证了 RSS 阅读器中能读取到站点中最新的数据。

问题 226　如何动态创建 XML 文件？

问题阐述

通过 XML 可以将数据存储在一个文档中，并且可以实现数据格式的转换和传递。那么如何在 PHP 中应用 XML 文档？如何通过 PHP 动态创建 XML 呢？

专家解答

在 PHP 中动态创建 XML 文档的方法有很多，下面讲解如何通过 DOM 对象动态创建 XML 文件。通过 DOM 类库创建 XML 文档应用到的方法如下：

（1）通过 new DomDocument()实例化一个 DomDocument 对象。语法如下：

```
$dom = new DomDocument('1.0','gb2312');              //创建 DOM 对象
```

（2）通过$dom->createElement 创建一个新元素节点。语法如下：

```
class DomDocument {
DomElement createElement ( string name [, string value] )
}
```

参数 name 是创建的元素的名称；参数 value 是创建元素的值。如果成功则返回一个元素节点对象，否则返回一个错误。

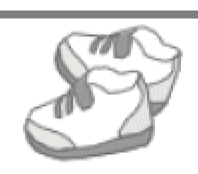
（3）通过$dom->appendChild 实现元素节点的附加。语法如下：

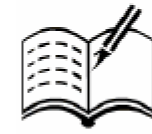
```
class DomNode {
DomNode appendChild ( DOMNode newnode )
}
```

参数 newnode 是将要附加的节点，返回更多的节点。

（4）通过$dom->createAttribute 创建一个新的属性。语法如下：

```
class DomDocument {
DomAttr createAttribute ( string name )
}
```





参数 name 是属性的名称。如果成功则返回一个属性对象，否则返回一个错误。

（5）通过$dom->createTextNode 创建一个属性值。语法如下：

```
class DomDocument {
DomText createTextNode ( string content )
}
```

参数 content 是属性的值。如果成功则返回一个属性值对象，否则返回一个错误。

（6）通过$dom->saveXML 生成一个 XML 文档。语法如下：

```
class DomDocument {
string saveXML ( [DOMNode node [, int options]] )
}
```

参数 node 指定生成的节点变量名称；参数 options 是附加的值。如果成功则返回一个 XML 文档，否则返回一个错误。

上述 6 个方法是在创建 XML 文档中需要使用的。

应用示例

下面应用上述的方法创建一个 XML 文档。其关键代码如下：

```
<?php
header('Content-Type:text/html;charset=utf-8');              //设置编码格式
$dom = new DomDocument('1.0','utf-8');                       //创建 DOM 对象
$object = $dom->createElement('object');                     //创建根节点 object
$dom->appendChild($object);                                  //将创建的根节点添加到 dom 对象中
$book = $dom->createElement('book');                         //创建节点 book
$object->appendChild($book);                                 //将节点 book 追加到 dom 对象中
$computerbook = $dom->createElement('computerbook');         //创建节点 computerbook
$book->appendChild($computerbook);                           //将 computerbook 追加到 dom 对象中
$type = $dom->createAttribute('type');                       //创建一个节点属性 type
$computerbook->appendChild($type);                           //将属性追加到 computerbook 元素后
$type_value = $dom->createTextNode('computer');              //创建一个属性值
$type->appendChild($type_value);                             //将属性值赋给 type
$bookname = $dom->createElement('bookname');                           //创建节点 bookname
$computerbook->appendChild($bookname);                                 //将节点追加到 dom 对象中
$bookname_value = $dom->createTextNode('学通 PHP24 堂课');             //创建元素值
$bookname->appendChild($bookname_value);                               //将值赋给节点 bookname
$xml_file=$dom->saveXML();
echo $xml_file;                                              //输出 XML 文件
file_put_contents('01.xml',$xml_file);                       //将 XML 文档保存到指定的文件中
?>
```

运行结果如图 11.3 所示。

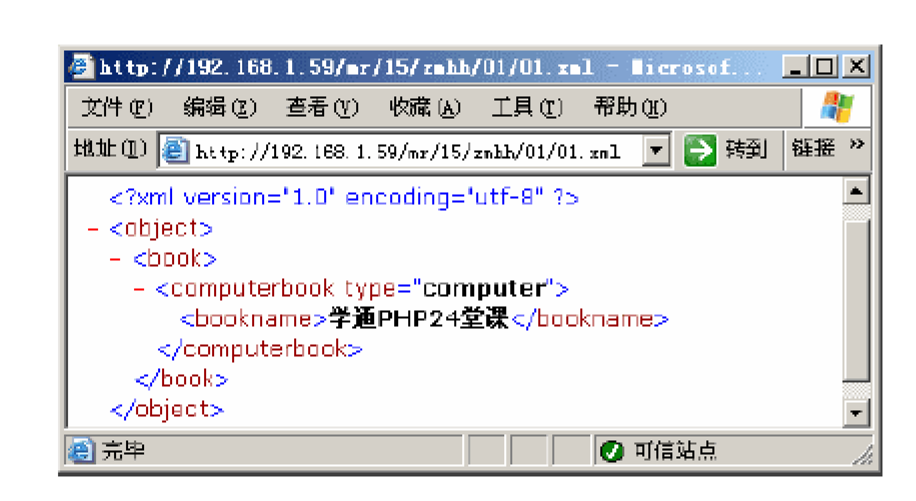
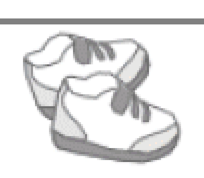


图 11.3　动态创建 XML 文档

专家点评

DOM（Document Object Model），文档对象模型，通过树型结构模式来遍历 XML 文档。使用 DOM 遍历文档的好处是不需要标记就可以显示全部内容，但缺点同样明显，就是非常消耗内存。



问题 227　如何动态读取 XML 文件？



问题阐述

既然通过 XML 可以将数据存储在一个文档中，那么如何读取这个文档中的数据呢？

专家解答

可以应用 DOM 类库中的 getElementsByTagName()方法动态读取文档中的数据。DOMDocument->getElementsByTagName()方法用于获取标签的名称，返回值为一个新的 DOMNodeList 对象。语法如下：

```
class DOMDocument {
    DOMNodeList getElementsByTagName ( string name )
}
```

参数 name 是 XML 文档中节点的名称。

应用示例

本示例应用 DOM 类库中的 getElementsByTagName()方法读取 XML 文档，并以 for 和 foreach 语句作为辅助，循环输出 XML 文档中的数据。本示例的所有功能都是在 index.php 文件中实现的，具体步骤如下：

（1）编写 PHP 脚本语句，创建一个 Read_XML()类，继承 DOMDocument 类库。在类文件中定义了两个函数：一个是构造函数，通过 load()方法读取存储在根目录下的 Rss.xml 文件；另一个是 show_xml()函数，应用 getElementsByTagName()方法读取 XML 文档中的数据，并通过 for 和 foreach 语句循环输出数据。其关键代码如下：

```
<?php
class Read_XML extends DOMDocument{                              //创建一个类
private $root;
public function __construct(){                                   //构造函数
     parent ::__construct();
     $this->load("Rss.xml");                                     //读取指定的 XML 文件
}
public function show_xml(){                                      //输出 XML 文件中的内容
     $root=$this->documentElement;                               //定义根目录
     $node_record=$this->getElementsByTagName("item");           //获取节点 item 下的标签
     $node_record_length=$node_record->length;                   //获取标签的数量
         for($i=0;$i<$node_record->length;$i++){                 //根据标签的数量执行 for 循环
          $k=0;
          foreach($node_record->item($i)->childNodes as $data){  //通过 foreach 语句输出标签中的数据
              $field[$k]= $data->textContent;
               $k++;
          }
          echo '<table width="672" border="1" cellpadding="1" cellspacing="1" bordercolor="#FFFFFF"
bgcolor="#999999"><tr>';
          echo '<td width="102" height="22" align="right" bgcolor="#FFFFFF">标题：</td>';
```



```
                    echo '<td width="368" bgcolor="#FFFFFF"> '.$field[0].'</td>';
                        echo '<td width="184" align="center" bgcolor="#FFFFFF">时间： '.$field[3].'</td>';
                        echo '</tr><tr>';
                    echo '<td colspan="3" height="40"; style="padding-left:5px; padding-right:5px; padding-top:5px;
padding-bottom:5px;" bgcolor="#FFFFFF">'.$field[2].'</td>';
                    echo '</tr><tr>';
                    echo '<td height="25" align="right" bgcolor="#FFFFFF">URL:</td>';
                      echo '<td colspan="2" bgcolor="#FFFFFF">  '.$field[1].'</td>';
                        echo '</tr></table>';
                    echo '</table>';
                }
            }
        }
        ?>
```

（2）创建 table 表格，添加查看 XML 文件的超链接。实例化 Read_XML 类，并执行 show_xml()方法，完成 XML 文件的读取。其关键代码如下：

```
        <table width="672" height="30" border="1" cellpadding="1" cellspacing="1" bordercolor="#FFFFFF"
bgcolor="#999999">
            <tr>
              <td height="30" align="center" bgcolor="#FFFFFF">PHP 读取 XML 文件   
     <a href="Rss.xml" class="STYLE3">查看 XML 文件内容</a></td>
            </tr>
            <?php
            $Read_XMl=new Read_XML;                                     //类的实例化
            $Read_XMl->show_xml();                                      //执行类中方法
            ?>
        </table>
```

（3）在示例的根目录中创建一个 Rss.xml 文件，并将其作为读取的内容。

通过 PHP 动态读取 XML 文档中的数据，其运行结果如图 11.4 所示。

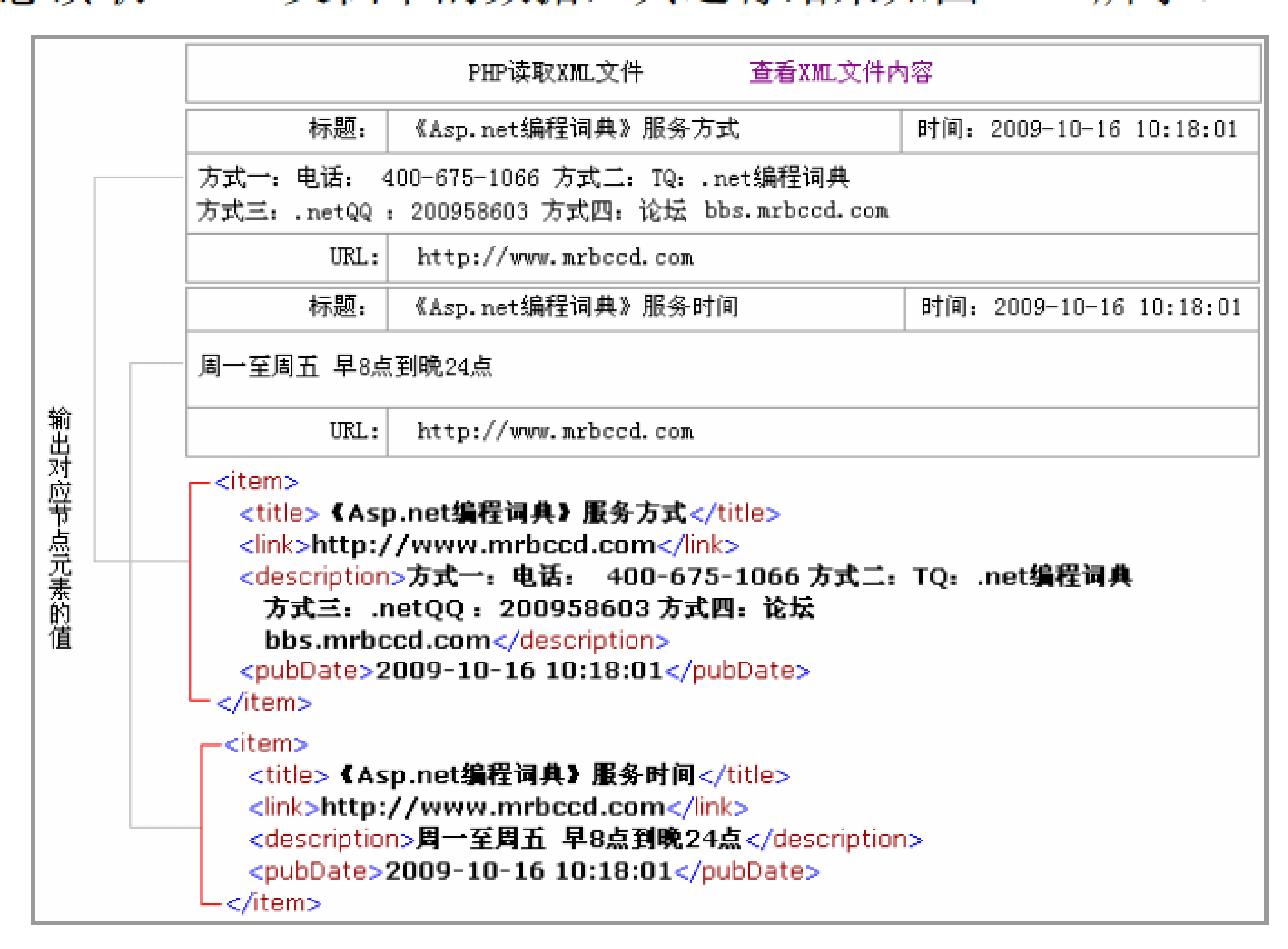
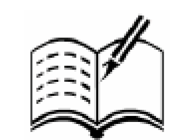
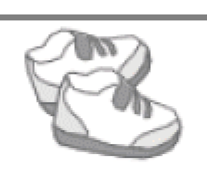
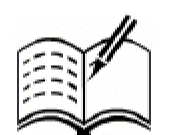


图 11.4 PHP 读取 XML 文件







Note

问题228　如何查询XML文件中的数据？

问题阐述

如何查询XML文件中的数据？

专家解答

在PHP中实现对XML的查询操作，应用的是DOM类库中的DOMXPath->query()方法。然后通过for和foreach语句循环输出查询结果。

（1）DOMXPath->query()方法用于查询出XML文件中符合条件的特定节点，返回值为一个节点列表对象。语法如下：

```
class DOMXPath {
    DOMNodeList query ( string expression [, DOMNode contextnode] )
}
```

其参数expression是执行的查询语句。若成功则返回一个节点列表对象DOMNodeList；若失败，则返回一个空的节点列表对象DOMNodeList。

本示例中使用的查询语句如下：

```
"//item[title='$search_title']"
```

查询出item节点中，title子节点的值等于变量$search_title的特定节点。

（2）$node_Record对象，查询返回的节点列表对象，通过for和foreach语句循环输出$node_Record对象中的数据。

DOMDocument->getElementsByTagName()方法用于获取标签的名称，返回值为一个新的DOMNodeList对象。语法如下：

```
class DOMDocument {
    DOMNodeList getElementsByTagName ( string name )
}
```

参数name是XML文档中节点的名称。

应用示例

通过DOM类库中的DOMXPath->query()方法查询XML文档中的数据，并且通过for和foreach语句循环输出查询结果。其具体步骤如下：

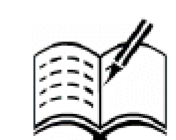
（1）通过jQuery技术动态输出整个XML文档中的数据。应用存储在js文件夹下的fun.js脚本文件和jquery-1.3.2.js框架包。

（2）创建index.php文件。首先，创建Search_XML类，继承DOM类库，定义构造方法读取指定的XML文件；定义Select_XML()方法，通过DOMXPath->query()方法完成对XML的查询操作。其关键代码如下：

```
<?php
class Search_XML extends DomDocument{          //创建类，继承DOM类库
    public function __construct() {            //构造方法，打开指定的XML文件
        parent:: __construct();
```



```
                $this->load("Rss.xml");
            }
            public function Select_XML($search_title){      //定义方法，执行查询操作，参数查询的关键字
                $xpath = new DOMXPath($this);              //应用 DOMXPath 方法进行查询
                $node_Record = $xpath->query("//item[title='$search_title']");      //执行查询
                    for($i=0;$i<$node_Record->length;$i++){                         //循环输出查询结果
                    $g=0;
                    foreach($node_Record->item($i)->childNodes as $node){       //读取数组中的数据
                        $field[$g]=$node->textContent;
                        $g++;
                    }
                    echo '<table width="672" border="1" cellpadding="1" cellspacing="1" bordercolor=
"#FFFFFF" bgcolor="#999999"><tr>';
                    echo '<td width="102" height="22" align="right" bgcolor="#FFFFFF">标题：</td>';
                    echo '<td width="368" bgcolor="#FFFFFF"> '.$field[0].'</td>';
                        echo '<td width="184" align="center" bgcolor="#FFFFFF">时间：'.$field[3].'</td>';
                        echo '</tr><tr>';
                    echo '<td colspan="3" height="40" style="padding-left:5px; padding-right:5px;
padding-top:5px; padding-bottom:5px;" bgcolor="#FFFFFF">  '.$field[2].'</td>';
                    echo '</tr><tr>';
                    echo '<td height="25" align="right" bgcolor="#FFFFFF">URL:</td>';
                      echo '<td colspan="2" bgcolor="#FFFFFF">  '.$field[1].'</td>';
                        echo '</tr></table>';
                    echo '</table>';
                }
            }
        }
        ?>
```



然后，通过 Script 脚本标记调用 fun.js 和 jquery-1.3.2.js 脚本文件，完成 XML 文件的读取。其关键代码如下：

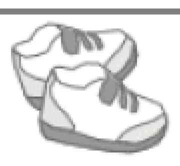
```
        <script src="js/jquery-1.3.2.js"></script>
        <script src="js/fun.js"></script>
```

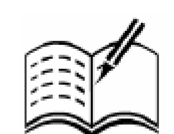
接着，创建 form 表单，添加表单元素，完成查询关键字的提交。其关键代码如下：

```
        <form name="form1" method="post" search_title="index.php">
        <tr>
            <td height=35 align="center" bgcolor="#CCCCCC"><span class="STYLE_select5">请输入标
题名称：</span>
                    <input name="search_title" type="text" id="search_title">
                <input type="submit" name="Submit" value="提交">
                <a href="Rss.xml" class="STYLE3">查看 XML 文件内容</a>
            </td>
        </tr>
        </form>
```

最后，实例化 Search_XML 类，执行 Select_XML()方法，将表单中提交的关键字作为参数传递到 Select_XML()方法中，完成对 XML 的查询。其关键代码如下：

```
        <?php
              if(isset($_POST['Submit']) && isset($_POST['search_title']) && $_POST['search_title']!=""){
```





```
                                                        //判断查询提交按钮是否为真
            $Search_XML = new Search_XML;               //实例化类
            $Search_XML->Select_XML($_POST['search_title']);  //执行类中方法，并对提交的
                                                         关键字进行编码转换
        }else{      //如果提交按钮和查询关键字为空，则输出如下内容
        ?>
        <div>
          <p> </p>
        </div>
        <?php
        }
?>
```

（3）在示例的根目录中创建一个 Rss.xml 文件，作为查询的内容。

运行结果如图 11.5 所示。

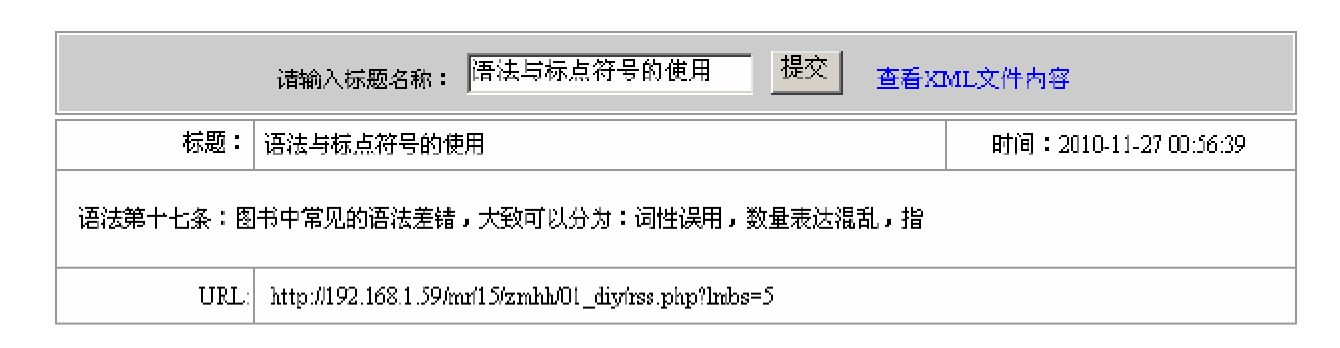


图 11.5　PHP 查询 XML 文件

问题 229　如何动态更新 XML 中的数据？

问题阐述

如何动态更新 XML 中的数据？

专家解答

要想实现动态更新 XML 中的数据，主要应用 DomDocument 类中的 createTextNode()和 replaceChild()方法完成对子节点值的修改。

（1）createTextNode()方法用于创建一个新的文本节点。语法如下：

```
class DOMDocument {
    DOMText createTextNode ( string content )
}
```

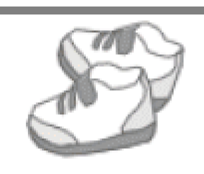
参数 content 是文本的内容。若成功，则返回一个文本节点；若失败，则返回一个错误信息。

说明：

TextNode()方法创建的文本节点不会出现在文档中，除非使用 DOMNode->appendChild()方法将其添加到文档中。

（2）replaceChild()方法用于修改一个子节点。语法如下：

```
class DOMNode {
    DOMNode replaceChild ( DOMNode newnode, DOMNode oldnode )
};
```





该方法实现将当前的 oldnode 节点替换为 newnode 节点。

应用示例

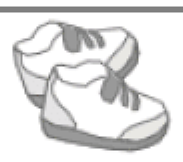
在 PHP 中修改 XML 文件，需要综合地运用读取和查询 XML 文件的技术。首先读取 XML 文件中的数据，然后根据超链接中传递的标题，通过查询定位到指定的 XML 文件，最终实现对 XML 文件的修改。其具体步骤如下：

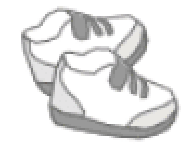
（1）完成类的创建和方法的定义。创建 Amend_XML 类，继承 DOM 类库。其中定义了 4 个方法。分别是构造函数，完成对指定 XML 文件的读取；show_amend()方法读取并输出 XML 文件中的数据；search_xml()方法执行查询操作，并将查询结果输出到 form 表单中，作为要修改的内容；amend_xml()方法完成对指定节点元素的修改操作。其关键代码如下：

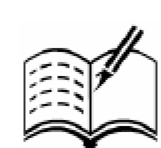
```
<?php
    class Amend_XML extends DomDocument{       //定义类文件，继承 DOM 类对象
    public function __construct() {            //构造函数
        parent:: __construct();
        $this->load("Rss.xml");                //读取存储留言信息的 XML 文档 Rss.xml
    }
//省略了部分代码
public function amend_xml($title,$link,$description){ //定义保存数据的方法
    $xpath = new DOMXPath($this);               //查询待修改的记录
    $Node_Record   = $xpath->query("//item[title=$title]");//执行查询操作，同样以标题为依据
    $Replace[0]=$title;
    $Replace[1]=$link;
    $Replace[2]=$description;
    $d=0;
    foreach ($Node_Record->item(0)->childNodes as $articles) {
        $Node_newText = $this->createTextNode($Replace[$d]);      //对提交的修改数据进行转码
        $articles->replaceChild($Node_newText,$articles->lastChild); //执行修改操作
        $d++;
    }
    echo "<script>alert('修改成功');location.href='".$_SERVER['PHP_SELF']."'</script>";
    $this->save("Rss.xml");
}
}
?>
```

（2）实例化 Amend_XML 类，应用 switch 语句，根据参数传递的不同值执行不同的方法，完成对 XML 文件的各种操作。其关键代码如下：

```
<?php
    $amend_xml = new Amend_XML;                 //实例化类
    $Action ="";                                //定义变量
    if(isset($_GET['Action']))                  //判断超链接传递的值
        $Action = $_GET['Action'];              //获取超链接传递的值
        switch($Action){                        //根据超链接传递的值进行判断
            case "show_amend":                  //浏览
                $amend_xml->show_amend();
            break;
            case "search_xml":                  //查询
                $amend_xml->search_xml($_GET['title']);
            break;
```







Note

```
                case "amend_xml":                          //修改
                    $amend_xml->amend_xml($_GET['title'],$_POST['link'],$_POST['description']);
                break;
                default:
                    $amend_xml->show_amend();
            }
?>
```

运行结果如图 11.6 所示。



图 11.6　PHP 修改 XML 文件

问题 230　如何动态删除 XML 中的数据？

问题阐述

如何动态删除 XML 中的数据？

专家解答

应用 DOM 类库中的 removeChild()方法删除 XML 文档中指定的子节点。removeChild()方法的语法如下：

```
class DOMNode {
    DOMNode removeChild ( DOMNode oldnode )
}
```

参数 oldnode 为被删除的子节点。若成功，则返回被删除的节点。删除成功后，需要通过 save()方法重新保存文件。

应用示例

在 PHP 中动态删除 XML 文件中的数据主要应用到 3 个自定义的方法：一是构造函数，完成读取指定 XML 文件的操作；二是 delete_xml()方法，执行 XML 的删除操作；三是 show_xml()方法，读取出 XML 文档中的数据。其具体步骤如下：

（1）在 index.php 文件中，创建 Delete_XML 类，继承 DOM 类库，定义构造方法完成对 XML 的读取；定义 delete_xml()方法执行删除 XML 节点的操作；定义 show_xml()方法实现输出 XML 文档中的数据。其关键代码如下：

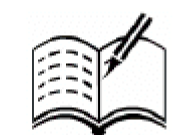
```
<?php
class Delete_XML extends DomDocument{     //定义类文件，继承 DOM
    private $Root;                        //定义成员变量
    public function __construct() {       //定义构造方法
        parent:: __construct();
```

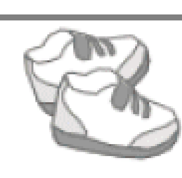


```
                $this->load("Rss.xml");           //读取 XML 文档 Rss.xml
            }
            public function delete_xml($title){   //定义删除 XML 节点的方法，以超链接传递的值为参数
                $Root=$this->documentElement;                        //定义根目录
                $xpath = new DOMXPath($this);                        //应用 XPath 方法查询数据
                $Node_Record= $xpath->query("//item[title='$title']"); //以标题为条件进行查询
                $Root->removeChild($Node_Record->item(0));           //根据查询结果，执行删除操作
                $this->save("Rss.xml");                              //重新保存文件
                echo "<script>alert('删除成功');location.href='".$_SERVER['PHP_SELF']."'</script>";
            }
            public function show_xml(){                                      //定义输出 XML 文档的方法
                $Node_Record = $this->getElementsByTagName("item");          //获取节点
                $Node_Record_Length    =$Node_Record->length;                //获取节点的个数
                for($i=0;$i<$Node_Record->length;$i++)  {                    //执行 for 循环
                    $K=0;
                    foreach ($Node_Record->item($i)->childNodes as $articles) {//应用 foreach 语句输出数据
                        $field[$K]=$articles->textContent;
                        $K++;
                    }
                echo '<table width="672" border="1" cellpadding="1" cellspacing="1" bordercolor="#FFFFFF"
bgcolor="#999999"><tr>';
                echo '<td width="102" height="22" align="right" bgcolor="#FFFFFF">标题：</td>';
                echo '<td width="368" bgcolor="#FFFFFF"> '.$field[0].'</td>';
                   echo '<td width="184" align="center" bgcolor="#FFFFFF"><a href="?Action=delete_
xml&title='.$field[0].'">删除</a></td>';
                   echo '</tr><tr>';
                echo '<td colspan="3" height="40" style="padding-left:5px; padding-right:5px; padding-top:5px;
padding-bottom:5px;" bgcolor="#FFFFFF">  '.$field[2].'</td>';
                echo '</tr><tr>';
                echo '<td height="25" align="right" bgcolor="#FFFFFF">URL:</td>';
                  echo '<td colspan="2" bgcolor="#FFFFFF">  '.$field[1].'</td>';
                   echo '</tr></table>';
                echo '</table>';
                }
            }
        }
        ?>
```

（2）实例化 Delete_XML，应用 switch 语句，根据传递的不同参数分别执行不同的方法，完成对 XML 的删除和输出操作。其关键代码如下：

```
        <?php
            $Action ="";                                        //定义变量
            $DeleteXML = new Delete_XML;                        //实例化删除类
            if(isset($_GET['Action']))                          //判断超链接传递的值
                $Action = $_GET['Action'];                      //获取超链接传递的值
                switch($Action){                                //根据超链接传递的值进行判断
                      case "delete_xml":                        //删除
                        $DeleteXML->delete_xml($_GET['title']); //执行删除操作
                    break;
                    default:
                    $DeleteXML->show_xml();
```

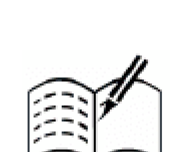




```
            }
    ?>
```

（3）在根目录下创建 Rss.xml 文件，定义被操作的内容。

运行本示例，结果如图 11.7 所示。单击结果页面中的“删除”超链接，即可删除指定的 XML 数据。



Note

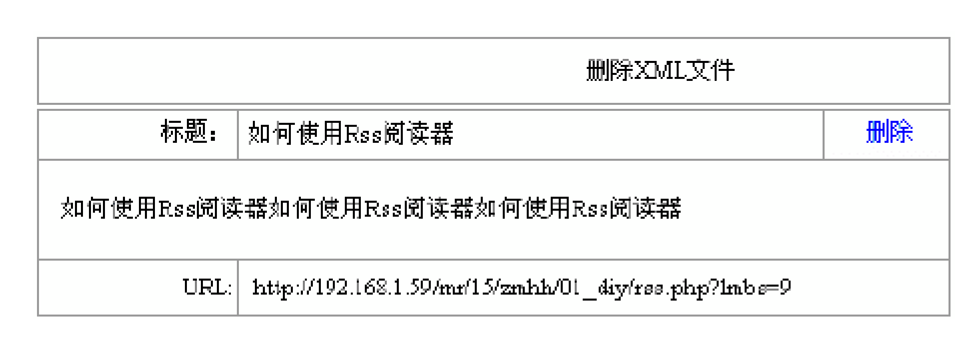


图 11.7　PHP 删除 XML 文件

问题 231　如何通过 jQuery 读取 XML 中的数据？

问题阐述

在提到读取 XML 文档中的数据时，大家可能都会想到 DOM 类库或 SimpleXML 对象，它们都可以完成对 XML 的读取操作，那么如何通过 jQuery 读取 XML 中的数据呢？

专家解答

通过 jQuery 读取 XML 中的数据应用到的技术如下。

（1）ready(fn)，当 DOM 载入就绪可以查询及操纵时绑定一个要执行的函数。通过 $(document).ready(fn)代替 body 标签的 onload 事件。

```
ready(fn)
```

参数 fn (Function)是在 DOM 就绪时执行的函数，可以为这个参数任意起一个名字。

（2）$.get()请求。语法如下：

```
jQuery.get(url,[data],[callback])
```

GET 请求返回一个 XMLHttpRequest。参数 url(String)是待载入页面的 URL 地址；data(Map)为可选参数，是待发送 Key/value 参数；callback (Function)也是可选参数，是载入成功时的回调函数。

（3）empty()删除匹配的元素集合中所有的子节点，其返回值是 jQuery。

（4）append(content)，向每个匹配的元素内部追加内容。这个操作与对指定的元素执行 appendChild()方法，将它们添加到文档中的情况类似。其返回值为 jQuery，参数 content (String, Element, jQuery)是要追加到目标中的内容。

（5）find(expr)搜索所有与指定表达式匹配的元素。该函数是找出正在处理的元素的后代元素的好方法。其返回值是 jQuery。参数 expr (String)是用于查找的表达式。

（6）each(callback)以每一个匹配的元素作为上下文来执行一个函数，即每次执行传递进来的函数时，函数中的 this 关键字都指向一个不同的 DOM 元素（每次都是一个不同的匹配元素），而且在每次执行函数时，都会给函数传递一个表示作为执行环境的元素在匹配的元素集合中所处位置的数字值作为参数（从零开始的整型）。其返回值是 jQuery。参数 callback (Function)



是对于每个匹配的元素所要执行的函数。如果返回 false，则停止循环（类似 break）；如果返回 true，则跳转至下一个循环（类似 continue）。

示例：迭代两个图像，并设置它们的 src 属性。代码如下：

```
<img/><img/>
$("img").each(function(i){
    this.src = "bgcolor" + i + ".gif";
 });
```

输出结果：

```
<img src=" bgcolor 0.gif" />, <img src=" bgcolor 1.gif" />
```

说明：

本示例中的 this 指的是 DOM 对象而不是 jQuery 对象

（7）text()取得所有匹配元素的内容。

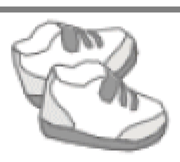
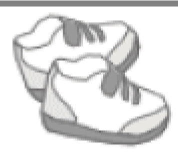
结果是由所有匹配元素包含的文本内容组合起来的文本。该方法对 HTML 和 XML 文档均有效。

应用示例

本示例将应用 jQuery 完成对 XML 的读取。其具体步骤如下：

（1）创建 index.php 文件，调用 jQuery 框架包，并且直接编写 JavaScript 脚本文件，应用 jQuery 中的函数完成对指定 XML 文件的读取，通过 setTimeout()函数每间隔 1000 毫秒执行一次函数，实现对 XML 文件内容的动态读取。其关键代码如下：

```
<script src="js/jquery-1.3.2.js" type="text/javascript"></script>
</head>
<script type="text/javascript">
$(document).ready(function ReadXMl(){
        $.get('Rss.xml', function(d){
                $('body').empty();                              //清空数据
                $('body').append('<h1> JQuery 动态读取 XMl 文件：<a href="Rss.xml" class="STYLE3">
查看 XML 文件内容</a></h1>');
                $('body').append('<dl />');                     //添加标签
                    $(d).find('item').each(function(){          //item 元素
                     var $item = $(this);
                     var title = $item.find('title').text();    //获取 title 的值
                     var links = $item.find('link').text();     //获取 link 的值
                     var description = $item.find('description').text();
                       var pubDate = $item.find('pubDate').text();
                     var html ='<table width="672" border="1" cellpadding="1" cellspacing="1" bordercolor=
"#FFFFFF" bgcolor="#999999"><tr>';
                          html +='<td width="102" height="22" align="right" bgcolor="#FFFFFF">标题：</td>';
                          html +='<td width="368" bgcolor="#FFFFFF"> '+title+'</td>';
                              html  +=  '<td  width="184"  align="center"  bgcolor="#FFFFFF">时 间：
'+pubDate+'</td>';
                              html += '</tr><tr>';
                          html  +=  '<td  colspan="3"  height="40";  bgcolor="#FFFFFF">  '+
description+'</td>';
                          html += '</tr><tr>';
```

```
                html += '<td height="25" align="right" bgcolor="#FFFFFF">URL:</td>';
                html += '<td colspan="2" bgcolor="#FFFFFF">  '+links+'</td>';
                html += '</tr></table>';
                 html +="</table>";
            $('dl').append($(html));                 //向 dl 中添加数据
        });
    });
    setTimeout("ReadXMl()", 1000);                   //间隔 1000 毫秒执行一次 ReadXML 函数
});
</script>
```

（2）创建 Rss.xml 文件，存储于示例的根目录下。

运行结果如图 11.8 所示。

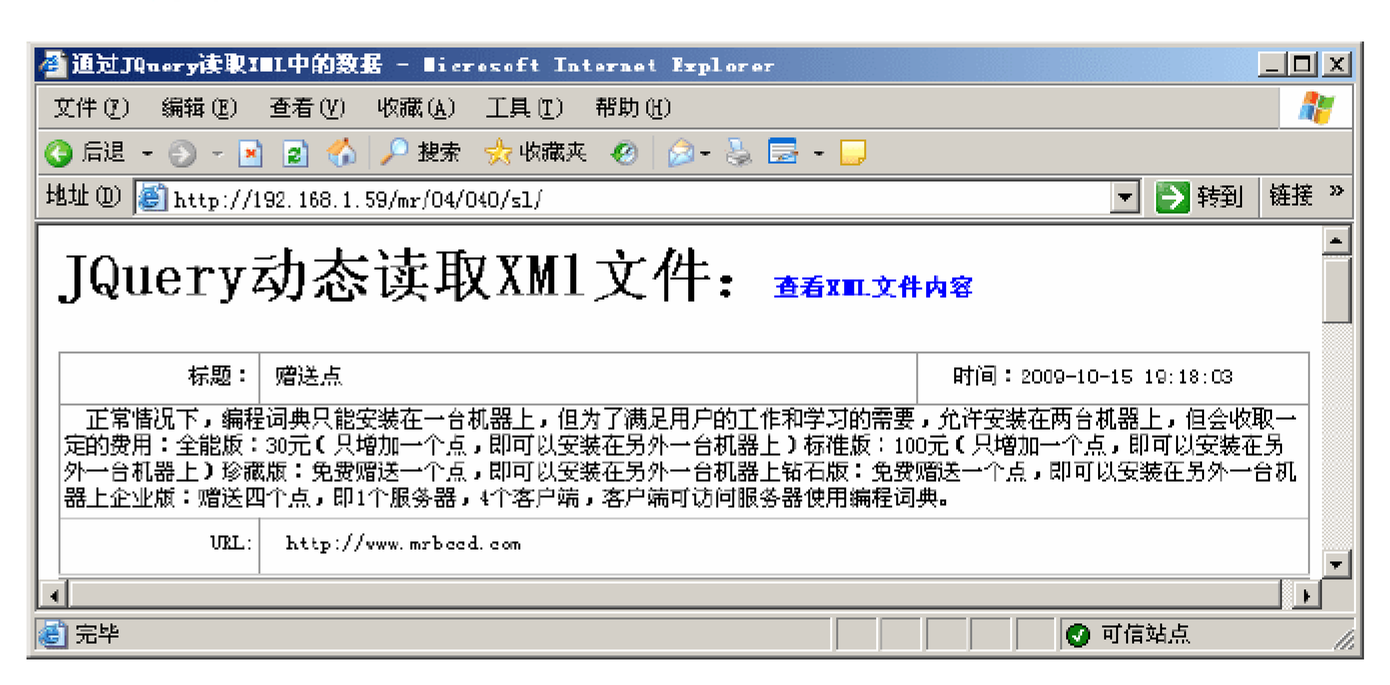


图 11.8　JQuery 动态读取 XML 文件

专家点评

在实现 XML 文件的动态读取时，为了能够获取到 XML 文件中最新的数据，必须对 Internet 的临时文件进行如图 11.9 所示的设置。其操作步骤是：在 IE 浏览器中选择“工具”→“Internet 选项”命令，在 Internet 临时文件中单击“设置”按钮，在弹出的设置对话框中选中“每次访问此页时检查”单选按钮，然后单击“确定”按钮。

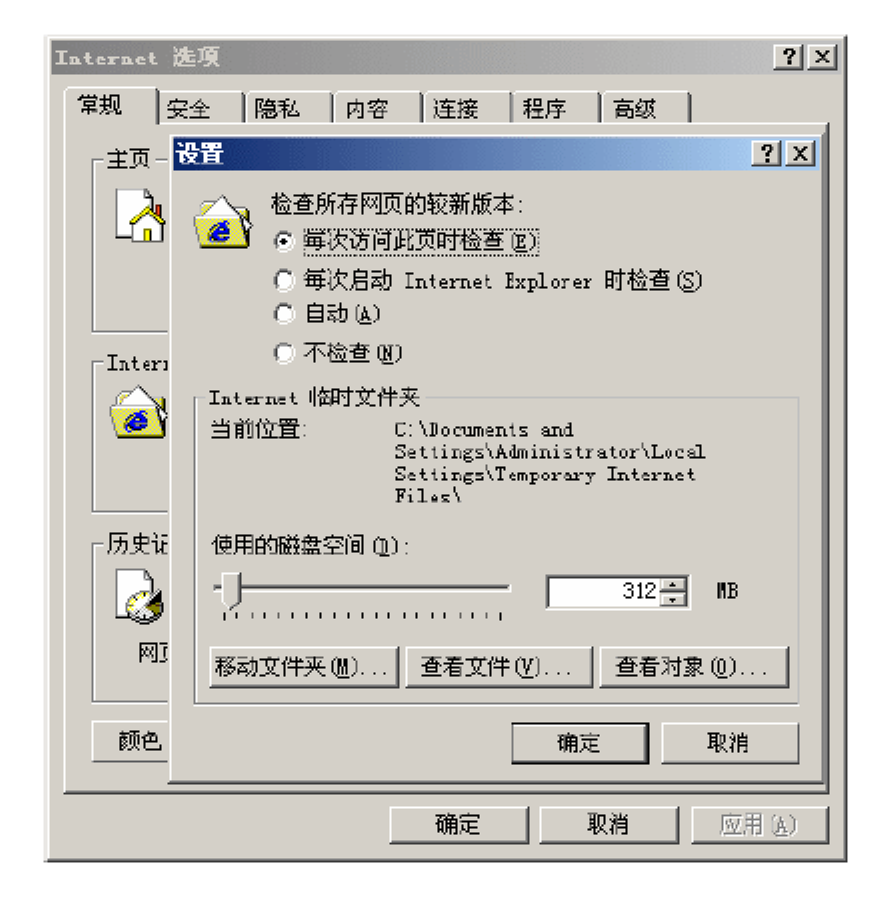
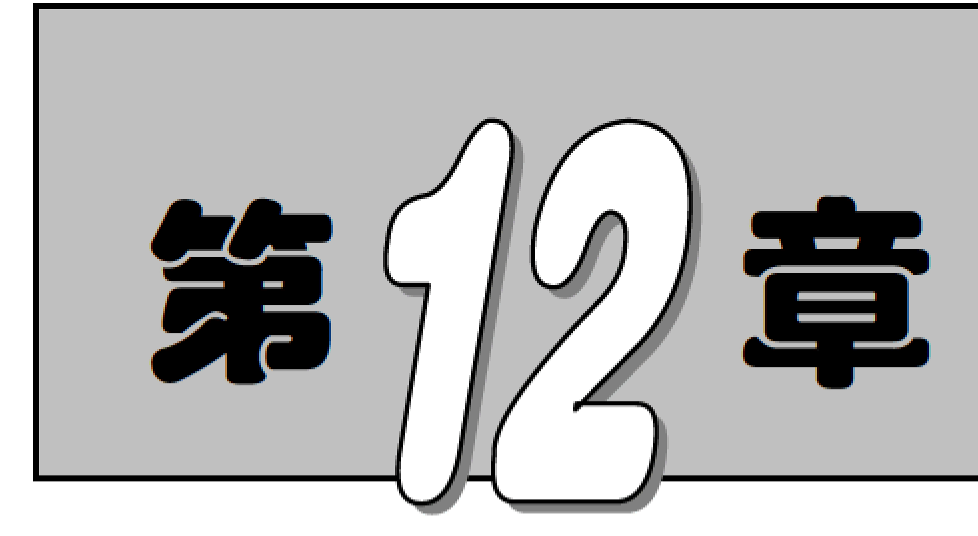


图 11.9　设置 Internet 临时文件

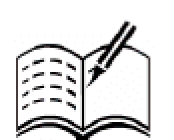


PDF 技术

- ⏭ 问题 232　如何创建简单的 PDF 文档?
- ⏭ 问题 233　如何在 PDF 中插入图片?
- ⏭ 问题 234　如何设置 PDF 文档的页眉与页脚?
- ⏭ 问题 235　如何在 PDF 文档中绘制表格?
- ⏭ 问题 236　如何在 PDF 文档中输出中文?
- ⏭ 问题 237　如何实现 PDF 文档下载?



问题 232　如何创建简单的 PDF 文档？



问题阐述

如何创建简单的 PDF 文档？

专家解答

通过 FPDF 可以创建 PDF 文档，其实现主要过程为：首先要实例化 FPDF 类，然后调用类中的函数（Open 函数、AddPage 函数、SetFont 函数、Cell 函数和 Output 函数）完成文档的创建。

（1）创建 FPDF 对象的语法格式如下：

```
$pdf = new FPDF([string page-orientation [, string measure-unit [, string page-format]]])
```

FPDF 对象的参数说明如表 12.1 所示。

表 12.1　FPDF 对象的参数说明

参　　数	说　　明
page-orientation	设置PDF文档的排版，可选值为：P：表示竖向；L：表示横向
measure-unit	设置文档中位置的计量单位。可选值为：pt：点；mm：毫米；cm：厘米；in：英寸
page-format	设置PDF文档纸张的类型。可选值：表示纸张类型的字符串，例如“A4”、“A5”、“Letter”，或者一个包含有两个元素的二维数组，直接指明纸张的大小

（2）Open 函数用于标示开始创建 PDF 文档，语法格式如下：

```
$pdf->Open()
```

（3）AddPage 函数为 PDF 文档添加一个新页，语法格式如下：

```
$pdf ->AddPage([string page-orientation])
```

其中，page-orientation 表示创建的 PDF 文档是横向还是竖向的。与前面介绍的创建对象时的使用方法相同。

（4）SetFont 函数设置当前使用的字体，语法格式如下：

```
$pdf->SetFont(string font [, string style [, float size]]);
```

其中，font 表示字体；style 表示样式，它的可选择值有如下 3 种（如果没有指定 style 的值，则默认为普通的形式）：

B：粗体

I：斜体

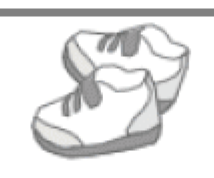
U：下滑线

size 表示字体的大小。如果不指定，则取默认值，为 12pt。

（5）Cell 函数用于为当前 PDF 文档增加一个单元格，其语法格式如下：

```
$pdf->Cell(float w [, float h [, string txt [, mixed border [, int ln [, string align [, int fill [, mixed link]]]]]]])
```

Cell 函数的参数说明如表 12.2 所示。



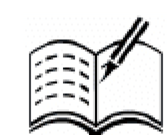


表 12.2　Cell 函数的参数说明

参　数	说　明
w	表示单元格宽度
h	表示单元格高度
txt	表示单元格中添加的文本字符串
bolder	表示单元格的边框，当设置为 0 时表示无边框，设置为 1 时表示有边框；或者 L 表示左边框，T 表示顶部边框，R 表示右边框，B 表示底部边框
ln	表示换行后，设置目前的位置。可选值为：0 表示向右移动；1 表示定位到下一行的开始；2 表示向下移动。默认值为 0
align	设置字符串的对齐方式。可选值为：L 表示左对齐；C 表示居中；R 表示右对齐。默认值为 L
fill	设置单元格的背景。可选值为：0 表示透明；1 表示不透明。默认值为 0
link	设置一个 URL 链接或由 AddLink 函数添加的内部链接

（6）Output 函数用于输出 PDF 文档，语法格式如下：

```
$pdf->Output([string filename [, bool download]])
```

其中，filename 表示要存储的文件名。如果不指定文件名，则浏览时会在 IE 浏览器中直接打开；download 表示是否一定要用户下载查看，默认为 False。

应用示例

下面应用上述介绍的函数创建一个 PDF 文档，输出一段英文字符串。

example1.php 的代码如下：

```
<?php
define('FPDF_FONTPATH','font/'); //定义 font 文件夹所在路径
require_once('pdf/fpdf.php');         //包含 FPDF 类库文件
$pdf=new FPDF('P', 'mm', 'A4');       //创建新的 FPDF 对象，竖向放纸，单位为毫米，纸张大小为 A4
$pdf->Open();                          //开始创建 PDF
$pdf->AddPage();                       //增加一页
$pdf->SetFont('Courier','I',20);       //设置字体样式
$pdf->Cell(0,0,'mrsoft!');             //增加一个单元格
$pdf->Output();                        //输出 PDF 到浏览器
?>
```

运行结果如图 12.1 所示。

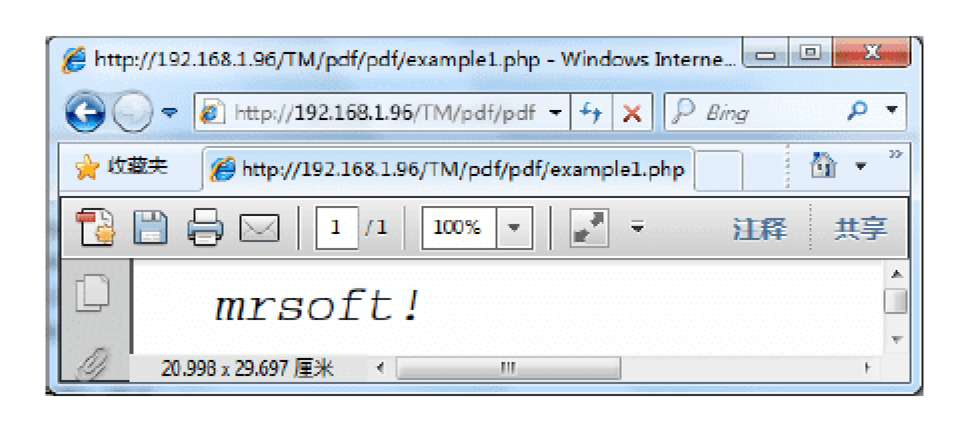


图 12.1　简单的 PDF 文档

技巧：

上面的示例是将 PDF 文档的内容在 IE 浏览器中输出，用户可以对上述代码中的“$pdf->Output();”进行修改，代码为“$pdf->Output('mrsoft.pdf',true);”，定义 PDF 文档名称 mrsoft.pdf，设置用户下载查看。如此即可将 PDF 文档保存到服务器并提示用户下载。





Note

专家点评

FPDF 是一个免费的 PDF 操作类，通过它可以完成基本的 PDF 创建功能，并且它还支持中文（需要对相应字体进行配置）。FPDF 类中常用的 PDF 操作函数包括：Open 函数、AddPage 函数、SetFont 函数、Cell 函数和 Output 函数。

FPDF 的下载可以通过访问 http://www.fpdf.org/来得到，下载完成后将下载的文件直接解压到某一文件夹中即可。

问题 233　如何在 PDF 中插入图片？

问题阐述

如何在 PDF 中插入图片？

专家解答

在 FPDF 中通过 Image 函数向 PDF 中插入图片，其语法格式如下：

```
Image(string file, float x, float y, float width, float height)
```

其中，x 和 y 表示图片所在的坐标，width 和 height 表示图片的宽度和高度。

应用示例

下面应用 Image()函数向 PDF 文档中插入一幅图片，并且将 PDF 在 IE 浏览器中直接输出。example2.php 的代码如下：

```
<?php
define('FPDF_FONTPATH','font/'); //定义 font 文件夹所在路径
require_once('pdf/fpdf.php');        //包含 fpdf 类库文件
$pdf=new FPDF('P', 'mm', 'A4');      //创建新的 FPDF 对象，竖向放纸，单位为毫米，纸张大小为 A4
$pdf->Open();                        //创建 PDF
$pdf->AddPage();                     //增加一页
$pdf->SetFont('Courier','I',20);     //设置字体样式
$pdf->Image('images/as.jpg',10,10,80,80);     //插入图片
$pdf->Output();                      //输出 PDF 到浏览器
?>
```

运行结果如图 12.2 所示。

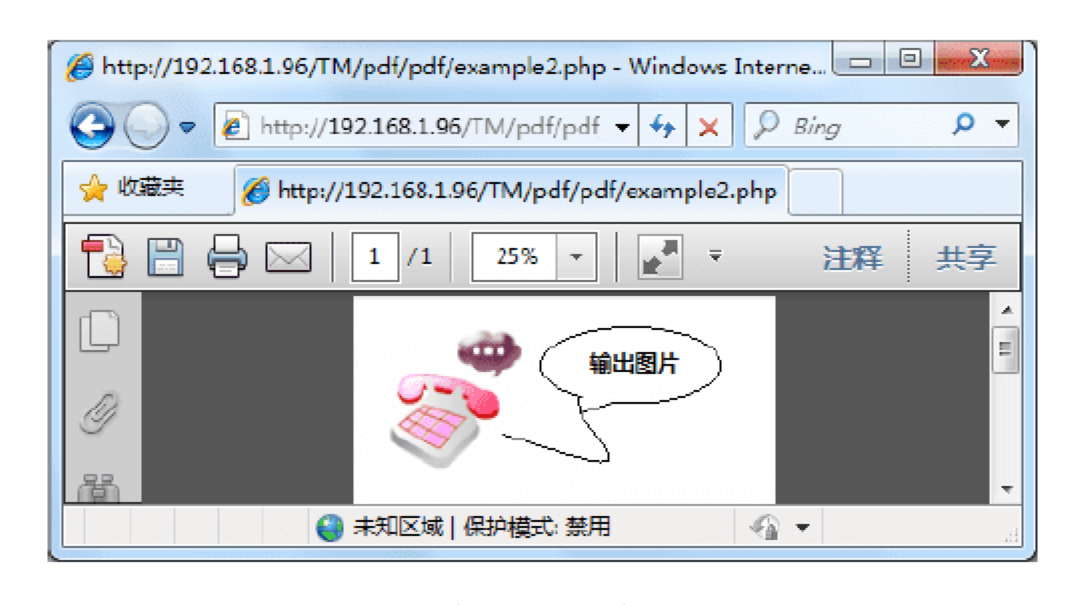
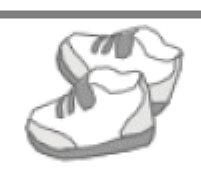
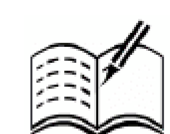


图 12.2　在 PDF 中插入图片





问题 234　如何设置 PDF 文档的页眉与页脚？

问题阐述

如何设置 PDF 文档的页眉与页脚？

专家解答

在 FPDF 中，页眉与页脚是通过重写 FPDF 类中的 Header 方法和 Footer 方法来实现的。因为 FPDF 类中的 Header 方法和 Footer 方法虽然存在，但是方法体中没有任何内容，所以设置 PDF 文档的页眉与页脚需要重写 FPDF 类中的 Header 方法和 Footer 方法。

应用示例

下面首先创建 PDF 类，继承 FPDF 类，在 PDF 类中对 Header 方法和 Footer 方法进行重写，完成页眉与页脚的设置。然后，创建 PDF 文档，输出一行英文字符串和一幅图片。

example3.php 的代码如下：

```
<?php
define('FPDF_FONTPATH','font/');                    //定义 font 文件夹所在路径
require_once('pdf/fpdf.php');                       //包含 FPDF 类库文件
class PDF extends FPDF{                             //继承 FPDF 类，重写页眉与页脚的方法
    function Header(){                              //设置页眉
        $this->SetFont('Arial','B',15);             //设置页眉字体
        $this->Write(10,'Title');                   //写入页眉文字
        $this->Ln(20);                              //换行
    }
    function Footer(){                              //设置页脚
        $this->SetY(-15);                           //设置页脚所在位置
        $this->SetFont('Arial','I',8);              //设置页脚字体
        $this->Cell(0,10,'Page - '.$this->PageNo()); //输出当前页码作为页脚内容
    }
}
$pdf=new PDF('P', 'mm', 'A4');       //创建新的 FPDF 对象，竖向放纸，单位为毫米，纸张大小为 A4
$pdf->Open();                        //创建 PDF
$pdf->AddPage();                     //增加一页
$pdf->SetFont('Courier','I',20);     //设置字体样式
$pdf->Cell(30,30,'MRSOFT!');         //输出字符串
$pdf->Image('images/as.jpg',30,50,80,80);      //插入图片
$pdf->Output();                                //输出 PDF 到浏览器
?>
```

运行结果如图 12.3 所示。

专家点评

在设置 PDF 的页眉与页脚时，可引入 Write、Ln、PageNo 和 SetY 4 个方法。

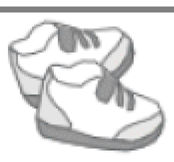
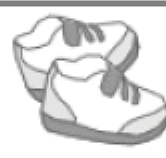




图 12.3　设置 PDF 的页眉和页脚

（1）Write 方法用于输出字符串。当到达文档的右边位置或者遇到（\n）时会自动换行。其语法格式如下：

```
Write(float h, string txt [, mixed link])
```

参数 h 定义字符串的行列高度；参数 txt 指定输出的字符串；参数 link 设置链接的网页或者 AddLink()方法的标识符。

（2）Ln 方法用于换行操作。其语法格式如下：

```
Ln([float h])
```

参数 h 用于设置行的高度。默认值为最后输出行的高度。

（3）PageNo 方法的作用是返回当前页码。其语法格式如下：

```
int PageNo();
```

（4）SetY 方法用于设置页脚的坐标。其语法格式如下：

```
SetY(float y);
```

参数 y 为页面上的 Y 坐标，单位为毫米。如果 y 为负数，则表示从页面底部向上的距离。例如，上面的示例中设置 y 的值为-15，即页脚所在位置距页面底部 15 毫米。

问题 235　如何在 PDF 文档中绘制表格?

问题阐述

如何在 PDF 文档中绘制表格?

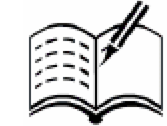
专家解答

通过 FPDF 绘制表格的方法与输出文字的方法类似，在输出文字时往往指定 Cell 方法的边框参数为 0，即不输出边框。在绘制表格时往往将其设置成大于 0 的整数，用于表示边框，这就是 PDF 文档中表格的单元格。当把这些单元格组合到一起后就构成了表格。

应用示例

下面在 PDF 中创建一个表格。example4.php 的代码如下：





```
<?php
define('FPDF_FONTPATH','font/');                    //定义 font 文件夹所在路径
require_once('pdf/fpdf.php');                       //包含 FPDF 类库文件
class PDF extends FPDF{
    function Header(){                              //设置页眉
        $this->SetFont('Arial','B',15);             //设置页眉字体
        $this->Write(10,'Pay Slip');                //写入页眉文字
        $this->Ln(20);                              //换行
    }
    function Footer(){                              //设置页脚
        $this->SetY(-15);                           //设置页脚所在位置
        $this->SetFont('Arial','I',8);              //设置页脚字体
        $this->Cell(0,10,'Page - '.$this->PageNo()); //输出当前页码作为页脚内容
    }
}
$pdf=new PDF('P', 'mm', 'A4');    //创建新的 FPDF 对象，竖向放纸，单位为毫米，纸张大小为 A4
$pdf->Open();                     //开始创建 PDF
$pdf->AddPage();                  //增加一页
$pdf->SetFont('Arial','',14);                       //设置字体样式
$header=array('Name','Age','Duty','Laborage');      //设置表头
$data=array();                                      //设置表体
$data[0] = array('Simon','24','employee','5,000.00');
$data[1] = array('Elaine','25','Manager','6,000.00');
$data[2] = array('Susan','25','employee','4,000.00');
$data[3] = array('David','26','Manager','8,000.00');
$width=array(40,40,40,40);                          //设置每列宽度
for($i=0;$i<count($header);$i++)                    //循环输出表头
$pdf->Cell($width[$i],6,$header[$i],1);
$pdf->Ln();
foreach($data as $row){                             //循环输出表体
    $pdf->Cell($width[0],6,$row[0],1);
    $pdf->Cell($width[1],6,$row[1],1);
    $pdf->Cell($width[2],6,$row[2],1);
    $pdf->Cell($width[3],6,$row[3],1);
    $pdf->Ln();
}
$pdf->Output();                                     //输出 PDF 到浏览器
?>
```

运行结果如图 12.4 所示。

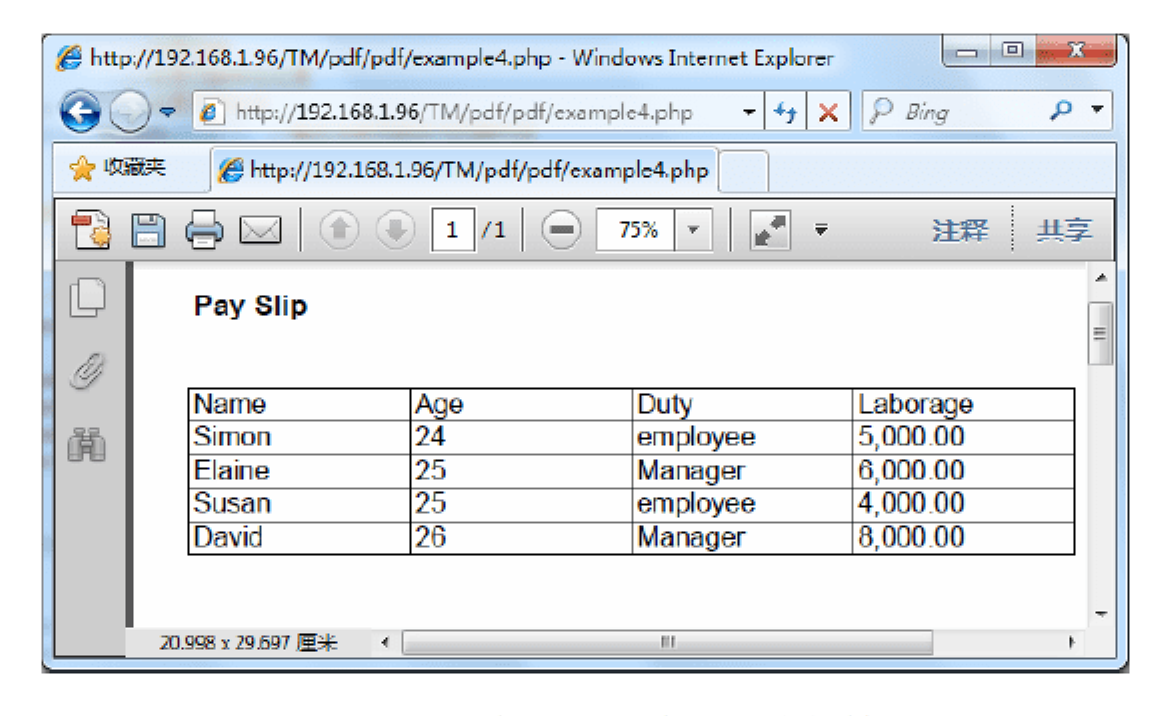
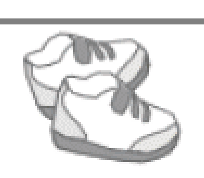
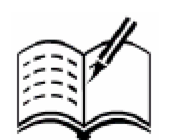


图 12.4　在 PDF 中绘制表格



问题 236　如何在 PDF 文档中输出中文？



问题阐述

若直接通过 FPDF 在 PDF 文档中输出中文，会发现有时输出的中文字符串是乱码。该如何解决这个问题？

专家解答

解决此问题的方法是应用 FPDF 提供的一个中文插件，继承插件中的 PDF_Chinese 类，在 PDF 文档中输出中文。

首先，下载并解压此插件，复制其中的 chinese.php 文件，并将其存储于项目的指定文件夹下，然后通过 include 语句在创建 PDF 文档的文件中载入 chinese.php，最后，创建 PDF 类，继承 PDF_Chinese 类，并在 PDF 文档中输出中文。

应用示例

本实例载入 PDF 中文插件，通过中文插件在 PDF 中输出中文字符串。

其实现的 example5.php 的代码如下：

```
<?php
include('pdf/chinese.php');                          //载入中文插件
class PDF extends PDF_Chinese{                       //继承中文插件类
    function Header(){                               //设置页眉
        $this->SetFont('GB','',10);                  //设置字体
        $this->Write(10,'FPDF 中文测试');            //设置页眉的内容
        $this->Ln(20);                               //执行换行操作
    }
    function Footer(){                               //设定页脚
        $this->SetY(-15);                            //设置页脚的输出坐标
        $this->SetFont('GB','',10);                  //设置字体
        $this->Cell(0,10,'第'.$this->PageNo().'页'); //设置页脚输出的内容
    }
}
$pdf=new PDF();                                      //创建 PDF 文档
$pdf->AddGBFont();                                   //添加字体
$pdf->Open();                                        //开启文档
$pdf->AliasNbPages();                                //为每个页面定义一个别名
$pdf->AddPage();                                     //添加页
$pdf->SetFont('GB','I',20);                          //设置字体
$str='山不在高，有仙则名。水不在深，有龙则灵。斯是陋室，惟吾德馨。苔痕上阶绿，草色入帘青。谈笑有鸿儒，往来无白丁。可以调素琴，阅金经。无丝竹之乱耳，无案牍之劳形。南阳诸葛庐，西蜀子云亭。孔子云："何陋之有？"';
$pdf->Write(10,$str);                                //输出中文
$pdf->Output();
?>
```



运行结果如图 12.5 所示。

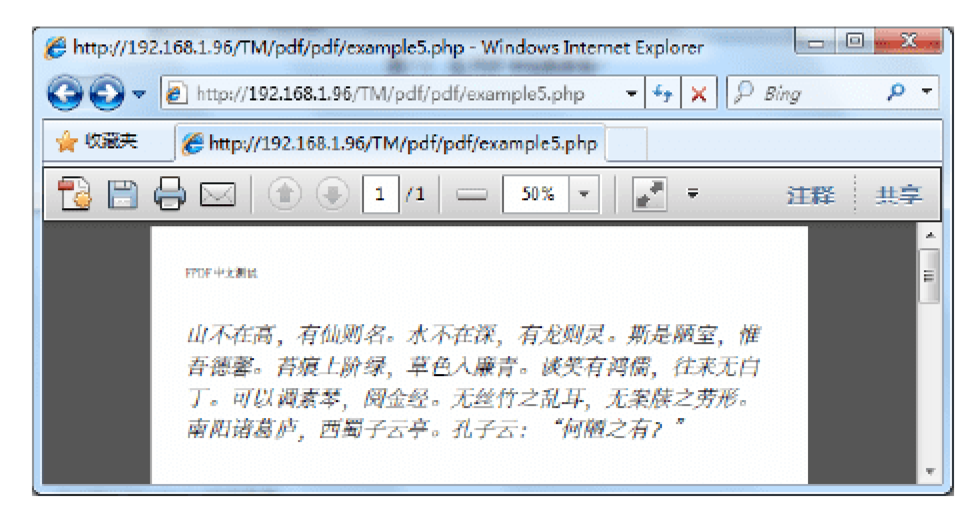


图 12.5　在 PDF 中输出中文



专家点评

在创建生成 PDF 文档的文件（example5.php）时，页面的编码格式应设置成 gb2312，如果页面的编码格式是 UTF-8，那么即使继承了 PDF_Chinese 类，输出的中文仍然会是乱码。

问题 237　如何实现 PDF 文档下载？

问题阐述

如何实现 PDF 文档下载？

专家解答

要实现 PDF 文档的下载操作，只需要在通过 Output 函数输出 PDF 文档时添加两个参数即可，第一个参数定义下载的 PDF 文档的名称，第二个参数设置为 true，提醒用户下载查看，如果将第二个参数设置为 false，那么生成的 PDF 文档会直接存储于服务器中，不会提醒用户下载查看。

应用示例

下面对示例 example5.php 进行修改，将其生成的 PDF 文档命名为“程序测试.pdf”，并且提醒用户下载查看。修改的关键代码如下：

```
<?php
include('pdf/chinese.php');                //载入中文插件
//省略了部分代码
$str='山不在高，有仙则名。水不在深，有龙则灵。斯是陋室，惟吾德馨。苔痕上阶绿，草色入廉青。谈笑有鸿儒，往来无白丁。可以调素琴，阅金经。无丝竹之乱耳，无案牍之劳形。南阳诸葛庐，西蜀子云亭。孔子云：“何陋之有？”  ';
$pdf->Write(10,$str);                      //输出中文
$pdf->Output('程序测试.pdf',true);
?>
```

修改后的示例存储于 example6.php 文件中，运行此示例将输出如图 12.6 所示的 PDF 文件下载页面。

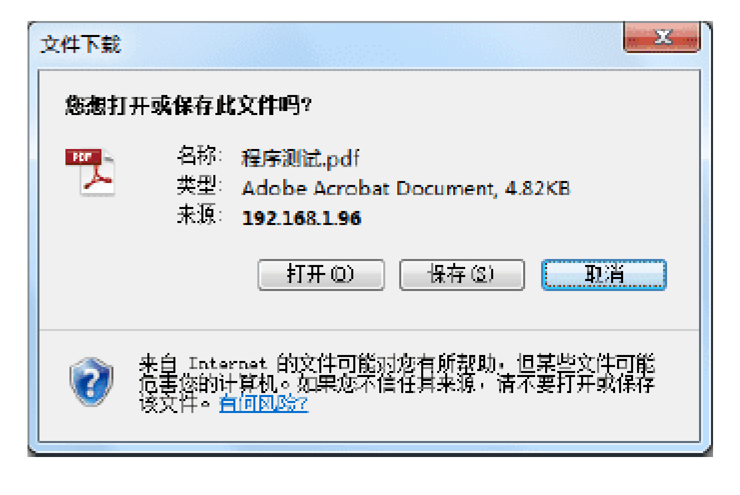


图 12.6　PDF 文档下载



PHP 与多媒体技术

- 问题 238　向 Flash 中传递参数
- 问题 239　如何通过 embed 标签播放音乐?
- 问题 240　如何通过 object 标签播放音乐?
- 问题 241　如何实现在线音乐中歌词同步输出?
- 问题 242　什么才是音乐在线播放的关键?
- 问题 243　（关闭网页时）Ajax 无刷新删除.m3u 格式的文件
- 问题 244　如何将选定的音乐添加到播放列表中?
- 问题 245　如何判断在线音乐文件的地址是否正确?
- 问题 246　如何在网页中播放 FLV 文件?
- 问题 247　如何实现 MP3 在线点播?





问题 238　向 Flash 中传递参数

问题阐述

在 Web 项目开发中，通过 Flash 动画来打造一些特殊的网页效果是很常见的方法。其中所用动画多数是单纯的动画效果，与项目中的数据联系不是很紧密，那么是否可以将动画中的内容与项目紧密地结合起来呢？根据动画中内容的变换，而改变项目中输出的内容呢？这就是我们将要解决的问题，根据 Flash 中传递的参数，输出不同的网页内容。

专家解答

向 Flash 中传递参数的具体步骤如下：

（1）根据项目中的需要创建 Flash，并且向 Flash 中传递参数值。在 Flash 元件中通过 on(release)方法获取传递的参数值，并执行超链接。

☑ 通过 this.loaderinfo.parameters["conn_type"]获取传递的参数值。

```
var type=this.loaderinfo.parameters["conn_type"];
```

Flash 的中的编辑结果如图 13.1 所示。

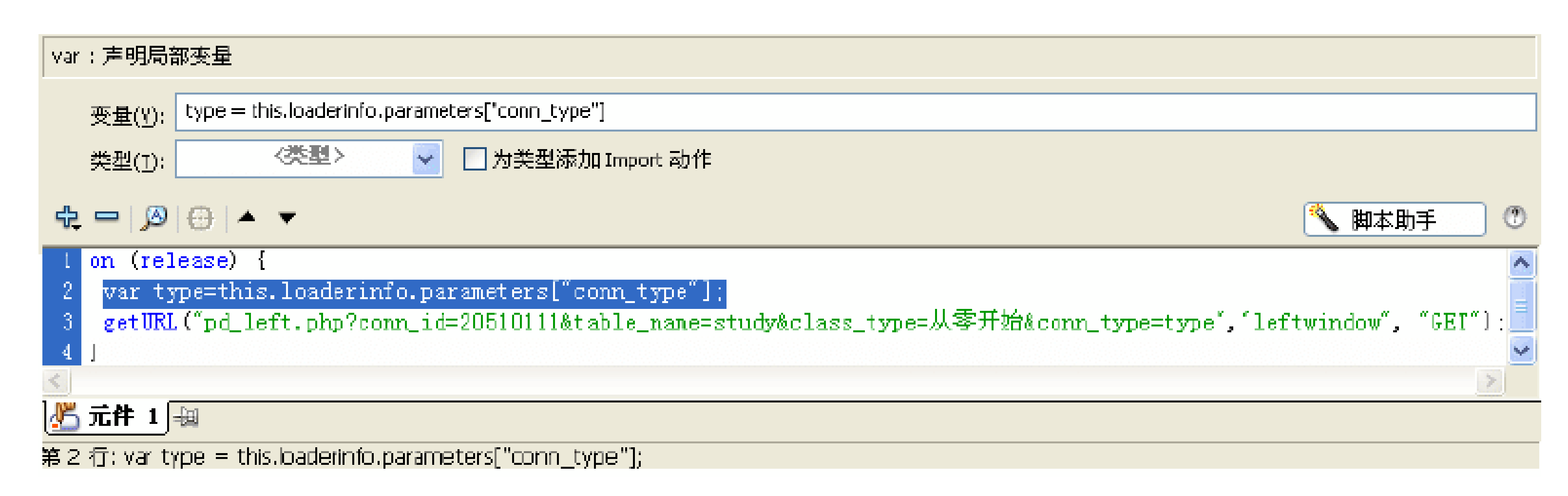


图 13.1　通过变量传递参数

☑ 通过 getURL 方法执行超链接。

```
getURL("pd_left.php?conn_id=20510111&table_name=study&class_type=从零开始&conn_type=type",
"leftwindow", "GET");
```

Flash 中的编辑结果如图 13.2 所示。

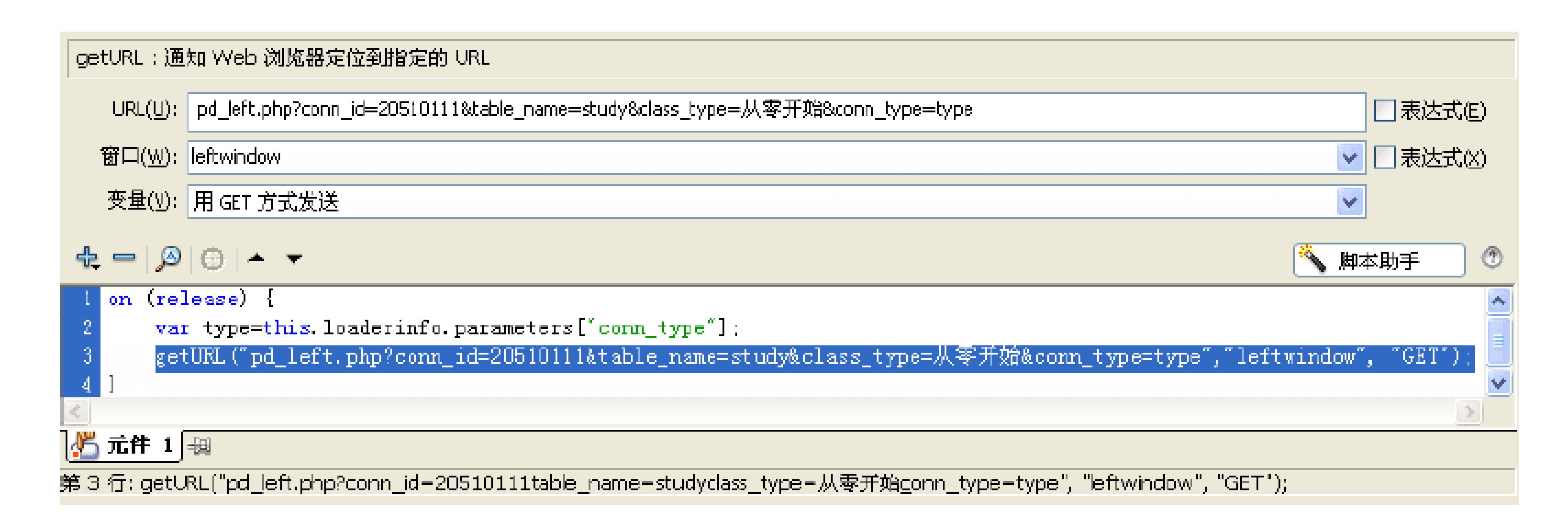
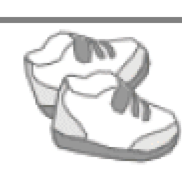
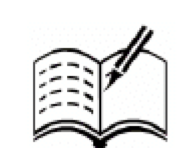


图 13.2　通过 getURL 执行超链接

此处在 Flash 中定义了 3 个元件，分别对应入门、开发和应用 3 个模块。在每个 Flash 元件中分别获取对应的参数值。





元件 1 的脚本代码如下：

```
on (release) {
    var type=this.loaderinfo.parameters["conn_type"];
    getURL("pd_left.php?conn_id=20510111&table_name=study&class_type=从零开始&conn_type=
type", "leftwindow", "GET");
}
```

元件 2 的脚本代码如下：

```
on (release) {
    var type=this.loaderinfo.parameters["conn_type"];
    getURL("pd_left.php?conn_id=10121002&table_name=jszx&class_type=技术中心&conn_type=
type", "leftwindow", "GET");
}
```

元件 3 中的脚本代码如下：

```
on (release) {
    var type=this.loaderinfo.parameters["conn_type"];
    getURL("pd_left.php?conn_id=10127105&table_name=ymzx&class_type=源码管理&conn_type=
type", "leftwindow", "GET");
}
```

（2）在项目中创建 js 脚本文件 flash.js，定义向 Flash 中写入的内容。其代码如下：

```
function flash(m){
    var  flash2='<object  classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"  codebase=
"http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,19,0" width="171" height="93">';
❶      flash2+='<param id="swf" name="movie" value="images/bccd.swf?conn_type='+m+'">';
       flash2+='<param name="quality" value="high">';
       flash2+='<param name="allScriptAccess" value="always"/>';
       flash2+='<param name="wmode" value="transparent">';
       flash2+='<embed src="images/bccd.swf" width="171" height="93" quality="high" pluginspage=
"http://www.macromedia.com/go/getflashplayer"  type="application/x-shockwave-flash"  wmode="transparent">
</embed>';
       flash2+='</object>';
❷      document.write(flash2);
}
```

代码导读：

❶ param id="swf"：在 swf 的 url 中传递参数值。

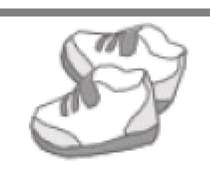
❷ document.write：执行写入操作。

（3）在项目的指定文件中载入定义的 js 脚本文件，调用定义的 flash()方法输出 Flash 动画。

```
<script src="js/flash.js"></script>
<script type="text/javascript">
    flash('{$conn_type}');
</script>
```

专家点评

在本程序的 LOG 中应用 Flash 实现入门、开发和应用 3 个模块之间的跳转，并根据 Flash





中传递的参数，在项目中完成不同内容之间的切换操作，其运行结果如图 13.3 所示。



图 13.3　通过 Flash 完成模块之间的跳转

问题 239　如何通过 embed 标签播放音乐？

问题阐述

在 Web 项目中，经常会看到支持在线音乐播放功能的网站，那么它们的音乐在线播放功能是如何实现的呢？这里我们将介绍众多方法中的一种，即通过 embed 标签实现音乐在线播放。

专家解答

<embed>是 html 标签中的播放多媒体标签，通过它可以实现音乐的在线播放功能。embed 标签的基本语法如下：

```
<embed src=url>
```

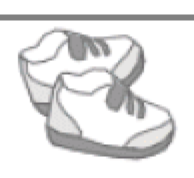
url：为音频或视频文件的路径，可以是相对路径，也可以是绝对路径。

embed 标签可以用来播放各种多媒体文件，格式可以是 midi、wav、mp3 等，目前主流的浏览器一般都支持该标签。

embed 标签有很多属性，常用的属性及说明如表 13.1 所示。

表 13.1　embed 标签常用属性及说明

参　数	说　明	举　例
autostart	该属性规定音频或视频文件是否在下载完之后就自动播放。当属性值为true时，音乐下载完成后自动播放；当属性值为false时，下载完成后不播放	<embed src="1.mp3" autostart=true>
loop	该属性规定音频或视频文件是否循环及循环次数。当属性值为true时，音频或视频文件循环；当属性值为false时，音频或视频文件不循环。如果为正整数，则为循环次数	<embed src="1.mp3" loop=true>
hidden	该属性规定控制面板是否显示，默认值为no。当属性值为true时，隐藏面板；当属性值为no时，显示面板	<embed src="1.mp3" hidden="yes">
starttime	该属性规定音频或视频文件开始播放的时间。如果未定义则从文件开头播放	<embed src="1.mp3" starttime="00:10">
volume	该属性规定音频或视频文件的音量大小。如果未定义则使用系统本身的设定	<embed src="1.mp3" volume="10">
width	该属性规定了控制面板的宽度	<embed src="1.mp3" width="100">
height	该属性规定了控制面板的高度	<embed src="1.mp3" height="200">
title	该属性规定音频或视频文件的说明文字	<embed src="1.mp3" title="a good song">



除了上述属性外，embed 标签还可以对面板的外观进行设置，包括面板的背景色和前景色等。

应用示例

播放存储于 upfile 文件夹下的.mp3 文件，其代码如下：

```
<embed src="upfile/brzy.mp3" align="middle" ShowStatusBar=true></embed>
```



Note

专家点评

通过 embed 标签不但可以实现音乐的在线播放，在充分发挥 loop 属性作用的前提下，还可以实现在线音乐的循环播放和连续播放。其具体原理是：通过 src 属性来获取播放文件的路径；判断当 loop 的属性值为 true 时执行循环播放；判断当 loop 的属性值为 false 时执行连续播放。例如，根据表单提交的值判断是执行循环播放还是连续播放。其关键代码如下：

```
<embed src="upfile/brzy.mp3" loop="<?php if($_POST['Submit']=="连续播放"){echo "false";} elseif($_POST['Submit2']=="循环播放"){echo "true";}?>" align="middle" ShowStatusBar=true></embed>
```

问题 240　如何通过 object 标签播放音乐？

问题阐述

能够实现音乐在线播放的方法众多，前面已经为读者介绍一种了，下面我们再介绍一种，即通过 object 标签实现音乐在线播放。

专家解答

object 标签定义一个嵌入的对象，通过该对象向 HTML 页面添加多媒体。object 标签的基本格式如下：

```
<object classid="clsid:22D6F312-B0F6-11D0-94AB-0080C74C7E95" height="68" id="MediaPlayer1" width="460">
    <param name="ShowStatusBar" value="-1">
    <param name="Filename" value="<?php echo $_GET[id]?>">
</object>
```

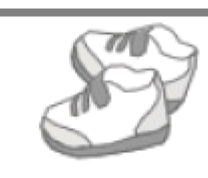
classid：设置或获取对象的类标识。
param：可定义用于对象的 run-time 设置。

应用示例

通过 object 标签播放指定的在线音乐文件，其关键代码如下：

```
<!--通过 object 标签播放指定的文件-->
<object classid="clsid:6BF52A52-394A-11D3-B153-00C04F79FAA6" id="mediaPlayer" width="480" height="64">
<param name="url" value="upfiles/video/<?php echo $_GET[id];?>">        //指定播放文件的路径
<param name="volume" value="100">
<param name="playcount" value="100">
<param name="enablecontextmenu" value="0">
<param name="enableerrordialogs" value="0">
</object>
```





专家点评

object 是微软专门为 IE 浏览器打造的、可以扩展外部应用程序及插件的对象标签，它和 embed 的不同之处在于：object 只支持以 IE 技术为核心的浏览器系列，而对其他的浏览无效，如果要使用 object 来播放多媒体，那么就需要安装相应的播放插件，如 realplay、quicktime 等。



问题 241　如何实现在线音乐中歌词同步输出？

问题阐述

大家经常会在在线音乐网站中看到在播放音乐的同时，对应的歌词也会同步输出，那么这个功能是如何实现的呢？这里我们就来探讨一下这个问题。

专家解答

对于歌词同步功能的实现，首先要解决的是歌词的问题，歌词文件以.lrc 为后缀。在歌词文件中存储的是在指定的时间段内所演唱的歌词。歌词文件中存储的内容如图 13.4 所示。

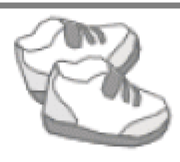
```
1215674743.lrc - 记事本
文件(F)  编辑(E)  格式(O)  查看(V)  帮助(H)
[ti:改变自己]
[ar:王力宏]
[al:改变自己]
[by:***王子]
[01:27.38]www.mrsoft.com  $$ ***王子 制作
[00:05.58]专辑:改变自己
[02:48.82][00:02.19]王力宏-改变自己
[00:14.81][00:00.47]
[00:00.62]COME ON
[00:09.65][00:01.56]
[02:42.09]编曲/电吉他:王力宏
[00:10.61]词/曲:王力宏
[00:19.65]今早起床了
[00:21.65]看镜子里的我
[00:24.03]忽然发现我发型
[00:26.32]睡的有点KUSO
[00:28.72]一点点改变
```

图 13.4　歌词文件中存储的内容

在拥有了歌词文件后，就可以通过 PHP 语句和 JavaScript 脚本语句实现歌词同步显示。其操作步骤如下：

（1）应用 Ajax 技术封装一个 JavaScript 方法 process()，实现歌词的无刷新读取。process()方法的语法如下：

```
//定义 XMLHttpRrequest 对象
var xmlHttp=createXmlHttpRequestObject();
//获取 XMLHttpRrequest 对象
function createXmlHttpRequestObject(){
    //用来存储将要使用的 XMLHttpRrequest 对象
    var xmlHttp;
    //如果在 Internet Explorer 下运行
    if(window.ActiveXObject){
        try{
            xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");
        }catch(e){
            xmlHttp=false;
        }
    }else{
    //如果在 Mozilla 或其他的浏览器下运行
        try{
            xmlHttp=new XMLHttpRequest();
        }catch(e){
            xmlHttp=false;
        }
    }
```



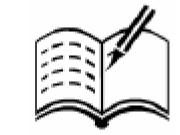

```
        //返回创建的对象或显示错误信息
        if(!xmlHttp)
            alert("返回创建的对象或显示错误信息");
            else
            return xmlHttp;
    }
    //使用 XMLHttpRequest 对象创建异步 HTTP 请求
    function process(ids){
        //在 xmlHttp 对象不忙时进行处理
        if(xmlHttp.readyState==4 || xmlHttp.readyState==0){
        xmlHttp.open("GET","lyric.php?id="+ids,true);
        //定义获取服务器端响应的方法
        xmlHttp.onreadystatechange=handleServerResponse;
        //向服务器发送请求
        xmlHttp.send(null);
        }else
        //如果服务器忙，1 秒后重试
        setTimeout('process()',1000);
    }
    //当收到服务器端的消息时自动执行
    function handleServerResponse(){
        //在处理结束时进入下一步操作
        if(xmlHttp.readystate==4){
            //状态为 200 表示处理成功结束
            if(xmlHttp.status==200){
                //获取服务器端发来的 XML 信息
                xmlResponse=xmlHttp.responseText;
                document.getElementById("lrcContent").innerHTML=xmlResponse;
            }else{
                //如果 HTTP 的状态不是 200 表示发生错误
                alert("There was a problem accessing the server:"+xmlHttp.statusText);
            }
        }
    }
```

（2）定义异步请求的 lyric.php 文件，应用 file_get_contents()函数读取歌词文件中的内容，在 js 的 handleServerResponse()方法中将歌词文件中返回的内容写入 ID 等于 lrcContent 的标签中。

（3）在音乐播放页中，在播放音乐的同时调用 process()方法读取歌词内容，在 div 标签中完成歌词的同步输出。其关键代码如下：

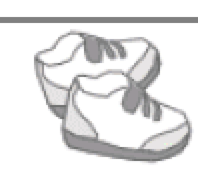
```
    <style>
    #lrcollbox td, #lrcollbox font {font-size: 16px;}
    #lrcoll td { color:#000000; }
    .lrcbc { overflow:hidden; height:20; width:0; filter:alpha(opacity=100); }
    </style>
    <script src="js/check_lyric.js" language="javascript"></script>
    <script>
        window.onunload=function(){
            clearTimeout(lrc0);
        }
```





```
        </script>
        <!--调用 process 函数，实现无刷新读取指定的歌词文件，并将内容输出到 lrcContent 中-->
        <body onLoad="process('brzy')" >
        <center>
        <span id="lrcContent" style="display:none;"><?php include("lyric.php");?></span>
        <object classid="clsid:6BF52A52-394A-11D3-B153-00C04F79FAA6" id="mediaPlayer" width="420"
height="64">
        <param name="url" value="upfiles/video/brzy.mp3">
        <param name="volume" value="100">
        <param name="playcount" value="100">
        <param name="enablecontextmenu" value="0">
        <param name="enableerrordialogs" value="0">
        </object>
        <div id="lrcollbox" style="overflow:hidden; height:100; width:420;">
        <table border="0" cellspacing="0" cellpadding="0" id="lrcoll" width="100%" style="position:relative;
top:120px;">
            <tr>
                <td nowrap height="20" align="center">
                    <table border="0" cellspacing="0" cellpadding="0">
                          <tr>
                            <td nowrap height="20"><span id="lrcbox1" style="height:20; color:#FF0000">
正在加载歌词……</span></td>
                          </tr>
                          <tr style="position:relative; top: -20px; z-index:6;">
                            <td nowrap height="20"><div id="lrcbc1" class="lrcbc"></div></td>
                          </tr>
                    </table>
                  </td>
            </tr>
        <?php
        $lyric=file_get_contents("upfiles/lyric/brzy.lrc");
        $counts=preg_match_all("/\[.*?\]/",$lyric,$tmptime); //提取[]，判断该行有几个时间
        for($i=0;$i<$counts;$i++){
        ?>
            <tr style="position:relative; top: <?php echo -20*$i;?>px;">
                <td nowrap height="20" align="center">
                    <table border="0" cellspacing="0" cellpadding="0">
                         <tr><td nowrap height="20"><span id="lrcbox<?php echo $i+2;?>" style=
"height:20"></span></td></tr>
                          <tr style="position:relative; top: -20px; z-index:6;"><td nowrap height="20"><div
id="lrcbc<?php echo $i+2;?>" class="lrcbc"></div></td></tr>
                    </table>
                  </td>
            </tr>
        <?php
        }
        ?>
        </table>
        </div>
```

（3）封装 JavaScript 脚本，实现歌词的滚动输出，其中涉及的 JavaScript 代码可参考本书



配套资源中的内容。

专家点评



歌词的同步输出关键在于 Ajax 无刷新技术的应用，通过它读取.lrc 格式的歌词文件。注意，在读取文本文件中的歌词时应用的是 file_get_contents()函数，而这个函数返回的字符串是 gb2312 编码格式，但是 Ajax 中默认使用的是 UTF-8 编码格式，所以在输出歌词时需要应用 iconv()函数对歌词的编码进行格式转换，只有将其转换为 utf-8 编码格式后才能够正常输出。

问题 242　什么才是音乐在线播放的关键？

问题阐述

音乐在线播放功能的实现，所使用的播放器、嵌入的多媒体标签，都不是最关键的内容，最为关键的是.m3u 格式文件的创建，通过它实现音乐的在线播放。

专家解答

在线播放功能的实现原理是客户端提交歌曲选项，服务器端生成.m3u 文件，并且将该文件通过 HTTP 协议下载到客户端，客户端调用相应的播放器在线播放音乐。

因此说创建一个.m3u 格式的文件是最关键的。创建.m3u 文件使用的是文件系统函数中的 fopen()和 fwrite()函数以及 foreach 语句。创建.m3u 文件的操作步骤如下：

（1）应用 fopen()函数以添加模式打开服务器中指定文件夹下的文件。如果在文件夹下不存在此文件，则创建一个新文件，文件的名称是当前时间的时间戳。代码如下：

```
<?php
$u=$_SERVER['HTTP_HOST'];                          //获取文件地址
$url="http://"."$u";                               //定义访问地址
$date=date("His");                                 //定义时间戳
$len=strripos($_SERVER['REQUEST_URI'],"/");
$tmp=substr($_SERVER['PHP_SELF'],0,$len);          //获取文件的根目录
if (!$fopen = fopen ("upfiles/video/$date.m3u","w")){ //使用添加模式打开文件，文件指针指在表尾
    echo "打开.m3u 文件失败！ " ;
    exit ;
}
```

（2）应用 foreach 循环语句，获取表单中提交歌曲的 ID 值。代码如下：

```
foreach($_POST[checkbox] as $key=>$value){ //以 book 数组做循环，输出键和值
```

（3）以获取到的歌曲 ID 值为查询条件，从数据库中读取出指定歌曲的存储名称。代码如下：

```
$l_sqlstr = "select id,style,name,actor,remark,address from tb_video where id='$value'order by id";
$l_rst = $conn->execute($l_sqlstr);                //以歌曲 ID 为查询条件进行查询
```

（4）将歌曲在服务器中存储的绝对路径写入.m3u 文件中，并且在绝对路径后添加一个“\n\r”分隔符。代码如下：

```
    if(!fwrite ($fopen,"".$url.$tmp."/".$l_rst->fields[5]."")){
        print "写入内容失败！ " ;
```



```
                    exit ;
            }
            if (!fwrite ($fopen,"\n\r")){                           //添加不同歌曲绝对路径之间的分隔符
                    print "写入内容失败！ " ;
                    exit ;
            }
    }
    ?>
```



至此，.m3u 文件创建完成，接下来就可以通过 embed 标签来读取.m3u 文件中存储的歌曲了。

```
        <embed src="upfiles/video/<?php echo $date;?>.m3u" loop="<?php if($_POST[Submit]=="连续播放")
{echo  "false";}elseif($_POST[Submit2]=="循环播放"){echo  "true";}?>"  align="middle"  ShowStatusBar=
true></embed>
```

专家点评

在线音乐播放时创建的.m3u 文件存储于服务器中指定的目录下，如果在网站中同时有很多人听音乐，那么就会创建很多的.m3u 文件，当这些人离开后，如果.m3u 文件仍然留在服务器的指定目录下，会占用大量的服务器空间，最终导致服务器空间被耗尽。因此必须在听歌的用户离开之后直接删除.m3u 文件，这样就避免了服务器空间的不必要的浪费。

问题 243　（关闭网页时）Ajax 无刷新删除.m3u 格式的文件

问题阐述

既然已经意识到.m3u 格式文件会对服务器构成威胁，那么我们就必须找出一个解决的方案，如何快速有效地在浏览者离开网站之后直接删除.m3u 格式的文件，这就是我们这里要解决的问题。

专家解答

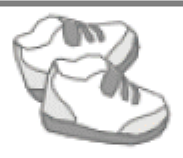
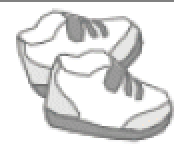
删除.m3u 文件是通过 onUnload 事件调用自定义函数 channel_check()完成的。在自定义函数 channel_check()中，应用 Ajax 无刷新技术调用 delete_m3u.php 文件实现删除.m3u 文件的操作。其操作的步骤如下：

（1）在播放页面的 body 标签中，通过 onUnload 事件调用 channel_check()函数，参数值是.m3u 文件在服务器中的存储位置。

```
    <body onUnload="channel_check('<?php echo "upfiles/video/".$date.".m3u"; ?>')">
```

（2）创建 channel_check()自定义函数，应用 Ajax 无刷新技术，调用 delete_m3u.php 文件，实现删除.m3u 文件的操作。其关键代码如下：

```
    function channel_check(names){
            if(xmlHttp.readyState==4 || xmlHttp.readyState==0){      //在 xmlHttp 对象不忙时进行处理
                xmlHttp.open("GET","delete_m3u.php?channel_name="+names,true);
                xmlHttp.onreadystatechange=handleServerResponse;//定义获取服务器端响应的方法
```

```
            xmlHttp.send(null);                          //向服务器发送请求
            return false;
        }else
        setTimeout('channel_check()',1000);              //如果服务器忙，1秒后重试
    }
    function handleServerResponse(){                     //当收到服务器端的消息时自动执行
        if(xmlHttp.readystate==4){
            if(xmlHttp.status==200){                     //状态为200表示处理成功结束
                xmlResponse=xmlHttp.responseText;        //获取服务器端发来的XML信息
            }else{
                alert("There was a problem accessing the server:"+xmlHttp.statusText);
            }
        }
    }
```

（3）在 delete_m3u.php 文件中，通过 unlink()函数删除服务器中指定文件夹下的.m3u 文件。代码如下：

```
<?php
header('Content-Type: text/html; charset=utf-8' );
//根据从客户端获取的用户创建输出
unlink($_GET[channel_name]);                             //删除指定的.m3u 文件
?>
```

专家点评

.m3u 文件不是音频文件，它是音频文件的列表文件。打开它，播放软件并不会播放它，而是根据它的记录找到网络地址进行在线播放。.m3u 文件很小，因为它里面没有任何音频数据。.m3u 格式的文件只是一个目录文件，提供了一个指向其他位置的音频视频文件的索引，播放的还是那些被指向的文件，用记事本打开.m3u 文件可以查看所指向文件的地址及文件的属性，以选用合适的播放器播放。

问题 244　如何将选定的音乐添加到播放列表中？

问题阐述

在线音乐网站中可以一次性选择多个要播放的音乐，并将其添加到播放列表中，可以选择循环播放，也可以选择连续播放。那么这个功能是如何实现的呢？

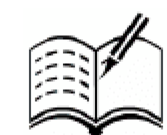
专家解答

播放列表类似于电子商务网站中的购物车，为用户选定的歌曲提供一个临时存放的空间。实现的原理是将用户选定歌曲的 ID 值存储到 SESSION 变量中，在播放列表中输出 SESSION 变量中的数据，重新读取歌曲的信息，播放歌曲。

首先，要将提交的音乐 ID 存储到 SESSION 变量中，并且判断 SESSION 中是否已经存在该音乐。其具体方法如下：

```
<?php
    session_start();
```



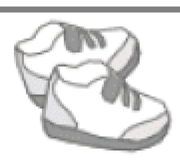


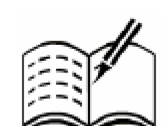

```
        include "conn/conn.php";
        $u=$_SERVER['HTTP_HOST'];
        $url="http://"."$u";

    if($_POST['Submit3']=="添加到播放列表"){
        if($_POST['checkbox']==""){
            echo "<script>alert('您没有选择要添加的音乐！'); history.back();</script>";
        }else{
            foreach($_POST[checkbox] as $key=>$value){   //以 book 数组做循环，输出键和值
                if($_SESSION["music_list"]==""){
                    $_SESSION["music_list"]=$value."@";
                }else{
                    $array=explode("@",$_SESSION["music_list"]);
                    if(in_array($value,$array)){
                            echo "<script>alert('播放列表中已经存在此音乐！');history.back();</script>";
                            exit;
                    }
                }
                    $_SESSION["music_list"].=$value."@";
            }
            echo "<script>window.location.href='music_list.php';</script>";
        }
    }else{
        if($_POST['checkbox']==""){
            echo "<script>alert('您没有选择要播放的音乐！'); history.back();</script>";
        }else{
            $date=date("His");
    ?>
```

然后，读取 SESSION 变量中存储的音乐 ID，根据该 ID 从数据库中查询出指定音乐文件的存储地址，并生成.m3u 文件，最终通过 embed 标签完成音乐的循环播放或连续播放。其方法应用如下：

```
    <center>
    <?php
    if(!$fopen = fopen ("upfiles/video/$date.m3u","w")){ //使用添加模式打开文件，文件指针指在表尾
        echo "打开.m3u 文件失败！";
        exit ;
    }
    $len=strripos($_SERVER['REQUEST_URI'],"/");
    $tmp=substr($_SERVER['PHP_SELF'],0,$len);        //获取文件的根目录
    foreach($_POST['checkbox'] as $key=>$value){      //以 book 数组做循环，输出键和值
        $l_sqlstr = "select id,style,name,actor,remark,address from tb_video where id='$value'order by id";
        $l_rst = $conn->execute($l_sqlstr);
        if(!fwrite ($fopen,"".$url.$tmp."/".$l_rst->fields[5]."")){
            print "写入内容失败！";
            exit ;
        }
        if(!fwrite ($fopen,"\n\r")){
            print "写入内容失败！";
```





Note

```
                exit ;
            }
        }
        ?>
        <embed src="upfiles/video/<?php echo $date;?>.m3u" loop="<?php if($_POST['Submit']=="连续播放"){echo "false";}elseif($_POST['Submit2']=="循环播放"){echo "true";}?>" align="middle" ShowStatusBar =true></embed>
        </center>
```

专家点评

在上述播放列表中，其核心是 SESSION 会话的应用，通过 SESSION 将数据临时存储于服务器中，使其可以在会话的有效时间内使用，当会话关闭之后，SESSION 变量的值将自动销毁，不会给服务器的运行带来过大的压力。

问题 245　如何判断在线音乐文件的地址是否正确？

问题阐述

在判断一个网站地址格式是否正确时，可以使用正则表达进行判断，也可以根据地址的类型和文件格式来进行判断。下面将探讨如何判断在线音乐文件的地址是否正确。

专家解答

判断在线音乐文件的地址是否正确，首先必须了解在线音乐文件的地址格式，例如 http://www.mrbccd.com/mp3/mrsoft.mp3，这就是一个名称为 mrosft.mp3 的音乐文件的访问地址，那么我们该如何判断这个地址的格式是否正确呢？

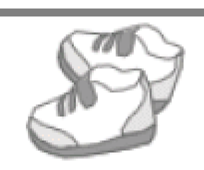
可以从两个方面入手，首先判断其是否是一个网址，即判断其是否以“http://”开头，然后判断音乐文件的后缀，可以是.mp3 或.avi 等多媒体格式文件。由此两处入手，基本上可以判断出文件地址是否正确。其具体方法如下：

封装 JS 脚本文件，通过 form1.internet_address.value.substr(0，7)！="http://"判断网址的开始位置是否符合 http://。其基本用法如下：

```
if(form1.internet_address.value.substr(0，7)!="http://"){
    alert("输入的网络地址的格式不正确!");
    form1.internet_address.select();
    return false;
}
```

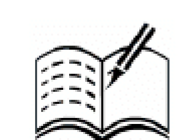
然后通过 form1.internet_address.value.substr(form1.internet_address.value.lastIndexOf('.'))判断网址中文件的后缀是否符合指定的要求。其基本用法如下：

```
//判断指定字符串中的格式符合.mp3
leng=form1.internet_address.value.substr(form1.internet_address.value.lastIndexOf('.'));
    if(leng!=".mp3" && leng!=".rm"&&leng!=".wav"&&leng!=".mid"){
        alert("输入的网络地址的格式不正确!");
        form1.internet_address.select();
```



```
                return false;
        }
```

只要提交的音乐文件的网址符合上述两个要求，那么它就是一个合格的在线音乐文件的地址。



专家点评

通过上述方法判断在线音乐文件的地址是否正确，只能说它基本符合了一个在线音乐文件地址的要求，至于这个地址具体是否存在，只能通过实践来检验了，而我们的判断只能对这个地址起到一个规范的作用。

问题 246　如何在网页中播放 FLV 文件？

问题阐述

如何在网页中播放 FLV 文件？

专家解答

要想在网页中播放 FLV 文件，首先应该在网页中嵌入 Flash 播放器，当该播放器嵌入成功后，应用该播放器播放 FLV 文件。

嵌入 Flash 播放器的方法有很多，这里应用 JS 嵌入 Flash 播放器。具体实现方法如下：

（1）下载播放器及其.js 文件。

（2）在 HTML 页面头部<head>区嵌入这个脚本文件：

```
<script type="text/javascript" src="swfobject.js"></script>
```

（3）在 HTML 中编写一个用来放 Flash 的容器，例如<div>，并设置其 ID，如 flash5。

```
<div id="flash5" class="right">FlvPlayer 播放器</div>
```

（4）编写 js 脚本，实例化 SWFObject 对象，载入 FlvPlayer 播放器，并且设置其中的各种参数，指定要播放的文件，其代码如下：

```
<script type="text/javascript">
    var s5 = new SWFObject("FlvPlayerV2009.swf","mediaplayer","500","320","8");
                                                    //实例化对象，设置播放器参数
    s5.addParam("allowfullscreen","true");
    s5.addVariable("width","500");                  //设置播放器宽度
    s5.addVariable("height","320");                 //设置播放器高度
    s5.addVariable("image","images/flash3.jpg");    //载入背景文件
    s5.addVariable("file","player.flv");            //指定播放文件
    s5.addVariable("backcolor","0xff8c00");         //背景颜色
s5.addVariable("frontcolor","0xE2F0FE");            //字体颜色
    s5.write("flash5");                             //写入文件
</script>
```

运行结果如图 13.5 所示。




Note



图 13.5 歌词文件中存储的内容

问题 247 如何实现 MP3 在线点播?

问题阐述

网上的很多音乐都提供在线收听和在线下载，那么如何才能实现 MP3 的在线点播呢?

专家解答

在网页中嵌入 MP3 播放器是实现 MP3 在线点播的关键，HTML 语言对该功能的实现提供了良好的支持，使用 HTML 语言中的<object>标签就可以调用系统自带的 Media Player 播放器，然后使用<param>标签传递需要播放文件的地址等参数，这样即可在网页中嵌入能够播放 mp3 音乐的播放器。<object>标签的基本语法如下：

```
<object classid="clsid:22D6F312-B0F6-11D0-94AB-0080C74C7E95" height="68" id="MediaPlayer1" width="460">
    <param name="AutoStart" value="-1">
    <param name="ShowStatusBar" value="-1">
    <param name="Filename" value="<?php echo $_GET[id]?>">
</object>
```

参数说明如下：

❶ AutoStart：该参数的值可以取 0 和-1，取-1 表示打开网页后自动播放，而取 0 表示只有单击播放器控制面板中的播放按钮才开始播放。

❷ ShowStatusBar：该参数的值可以取 0 和-1，取-1 表示播放器有状态栏，而取 0 表示播放器无状态栏。

❸ Filename：该参数用来指定播放文件的地址。

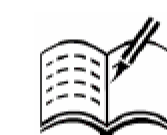
应用示例

本示例实现一个 MP3 文件在线点播的功能。其具体步骤如下：

（1）将<object>标签嵌入到网页中，代码如下：



```
    <object  classid="clsid:22D6F312-B0F6-11D0-94AB-0080C74C7E95"  height="68"  id="MediaPlayer1"
width="460">
        <param name="AutoStart" value="-1">
        <param name="ShowStatusBar" value="-1">
        <param name="Filename" value="<?php echo $_GET[id];?>">
    </object>
```



（2）通过单击“试听”文字超链接实现 MP3 的调用与播放，代码如下：

```
    <a href="index.php?id=mp3/music.mp3">试听</a>
```

运行本示例，单击所选歌曲名称后面的“试听”文字超链接，将弹出如图 13.6 所示的播放器，可以通过该播放器的控制面板对音乐进行播放控制。



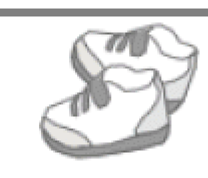
图 13.6　MP3 在线点播

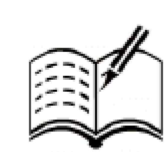
专家点评

Object 标签是用于 Windows IE3.0 及以后浏览器或者其他支持 Activex 控件的浏览器。如果用户没有安装过 flash player 那么 IE3.0 以后的浏览器会跳出一个提示框访问是否要自动安装 flash player。可以通过“classid”和“codebase”属性来告诉浏览器自动下载 flash player 的地址。



Smarty 模板





问题 248　如何安装和配置 Smarty 模板？

问题阐述

Smarty 模板引擎属于第三方的组件，要应用其实现网页的动静分离，首先就要从网站上下载该组件，并且将其放置到具体的项目中，在进行配置后才可以使用。那么如何下载、安装和配置 Smarty 模板呢？我们将在这里揭晓谜底。

专家解答

1. 安装 Smarty 模板

（1）将下载的 Smarty 压缩包解压后出现一个 libs 目录，这里包含了 Smarty 类库的核心文件，包括 smarty.class.php、smarty_Compiler.class.php、config_File.class.php 和 debug.tpl 4 个文件，还有 internals 和 plug-ins 两个目录，如图 14.1 所示。

（2）复制 libs 目录到服务器根目录指定的文件夹下。至此，Smarty 安装成功。

2. Smarty 模板引擎配置

Smarty 模板引擎的配置步骤如下：

（1）确定 Smarty 类库的存储位置。因为 Smarty 类库是通用的，每一个项目都可能会使用到它。所以将 Smarty 存储在根目录下是一个比较理想的选择，存储于根目录的 Smarty 文件夹下。

（2）在 Smarty 目录下新建 4 个目录 templates、templates_c、configs 和 cache，如图 14.2 所示。

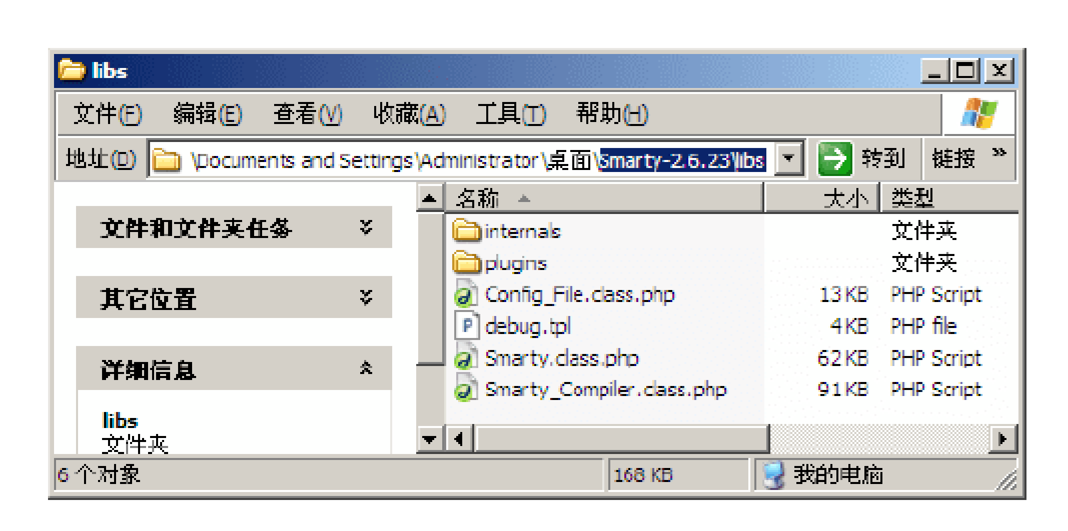


图 14.1　Smarty-2.6.23 文件夹

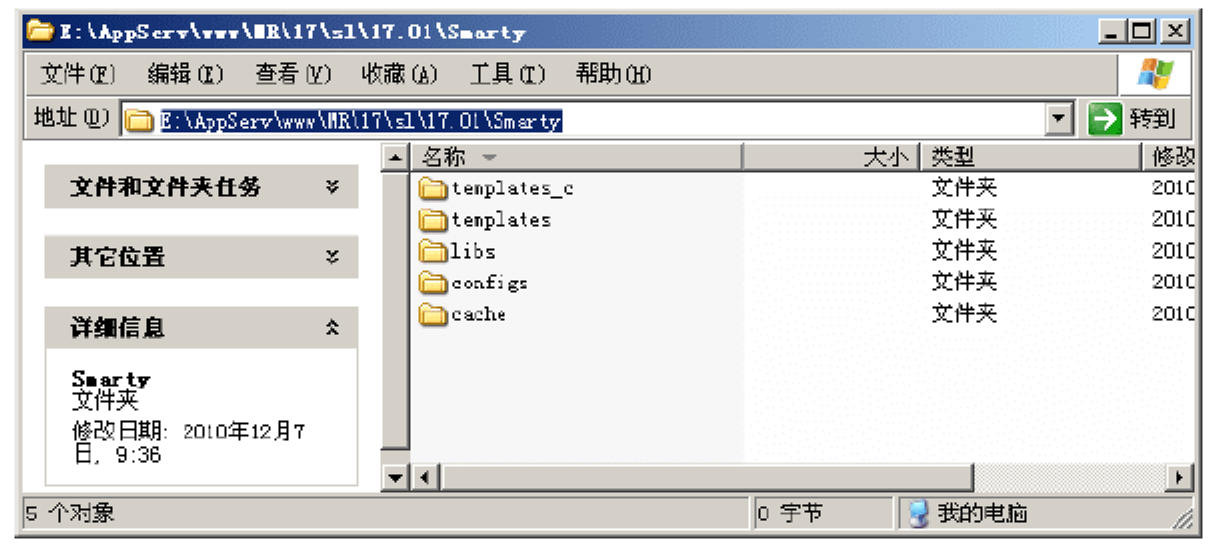


图 14.2　Smarty 目录中的内容

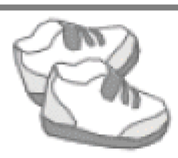
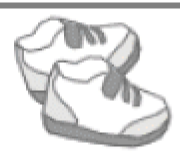
说明：

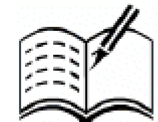
新建的 4 个目录并没有固定的存储位置，只要配置的路径正确，存储在根目录下的任何位置都可以。

（3）创建配置文件。当所有需要的资源都就绪之后，下面要做的就是将它们进行整合、调动起来。这就是配置文件的作用，定义服务器、Smarty 的绝对路径；加载 Smarty 类库文件；为创建的 4 个目录设置正确的路径；定义 Smarty 的定界符等操作都可以在配置文件中设置。

通常将配置文件定义到一个单独的文件中，在需要使用时直接通过包含语句调用即可。这里将配置文件存储到 config.php 中，其代码如下：

```
<?php
define('BASE_PATH',$_SERVER['DOCUMENT_ROOT']);            //定义服务器的绝对路径
```

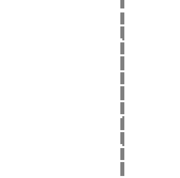





Note

```
define('SMARTY_PATH','\MR\14 \Smarty\\');                            //定义 Smarty 目录的绝对路径
require BASE_PATH.SMARTY_PATH.'libs\Smarty.class.php';               //加载 Smarty 类库文件
$smarty = new Smarty;                                                //实例化一个 Smarty 对象
$smarty->template_dir = BASE_PATH.SMARTY_PATH.'templates/';          //定义模板文件存储位置
$smarty->compile_dir = BASE_PATH.SMARTY_PATH.'templates_c/';//定义编译文件存储位置
$smarty->config_dir = BASE_PATH.SMARTY_PATH.'configs/';              //定义配置文件存储位置
$smarty->cache_dir = BASE_PATH.SMARTY_PATH.'cache/';                 //定义缓存文件存储位置
/*   定义定界符   */
$smarty->left_delimiter = '<{';
$smarty->right_delimiter = '}>';
?>.
```






上述配置文件的参数说明如下：

❶ BASE_PATH：指定服务器的绝对路径。

❷ SMARTY_PATH：指定 Smarty 目录的绝对路径。

❸ require()：加载 Smarty 类库文件 Smarty.class.php。

❹ $smarty：实例化 Smarty 对象。

❺ $smarty->template_dir：定义模板目录存储位置。

❻ $smarty-> compile_dir：定义编译目录存储位置。

❼ $smarty-> config_dir：定义配置文件存储位置。

❽ $smarty-> cache_dir：定义模板缓存目录。

❾ $smarty->left_delimiter：定义 Smarty 使用的开始定界符。

❿ $smarty->right_delimiter：定义 Smarty 使用的结束定界符。

说明：

关于定界符的使用，开发者可以指定任意的格式，也可以不指定。Smarty 默认的定界符是“{”和“}”。

至此，Smarty 的配置讲解完毕。至于将配置文件存储在什么位置，可以根据实际情况而定。

专家点评

Smarty 是一个使用 PHP 编写的应用于 PHP 的模板引擎，它将一个应用程序分成两部分：视图和逻辑控制。简单地讲，就是将 UI（用户界面）和 PHP Code（PHP 代码）分离。这样，程序员在修改程序时，就不会影响到页面设计，而美工在重新设计或修改页面时，也不会影响程序逻辑。

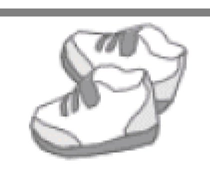
应用上述方法配置 Smarty 模板，当存储实例的文件夹发生变化时，就必须修改 config.php 配置文件中常量 SMARTY_PATH 的值，否则将无法找到 Smarty 类库的存储位置。

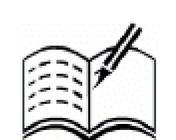
问题 249　如何将 Smarty 的配置方法封装到类中？

问题阐述

直接定义 Smarty 模板的配置方法，当存储实例的文件夹发生变化时，就必须修改 config.php 配置文件中常量 SMARTY_PATH 的值，否则将无法找到 Smarty 类库的存储位置。那么是否可以将 Smarty 模板的配置方法封装起来？这样无论实例的位置如何改变，都不会影响到 Smarty







的应用。

专家解答

将 Smarty 配置方法封装到类中，通过类中方法实现 Smarty 模板引擎中方法的调用，这样无论实例位置如何改变，都不会影响到 Smarty 的应用。Smarty 配置类的封装方法如下：

（1）创建 system.smarty.inc.php 文件。首先，包含 Smarty 类文件 Smarty.class.php。然后，定义 SmartyProject 类，继承 Smarty 父类。最后，定义 SmartyProject()方法，设置 Smarty 中模板文件（templates）、编译文件（templates_c）、配置文件（configs）和缓存文件（cache）的存储位置。配置文件 system.smarty.inc.php 的代码如下：

```
<?php
require("../Smarty/Smarty.class.php");                    //调用 Smarty 文件
class SmartyProject extends    Smarty{                    //定义类，继承 Smarty 父类
      function SmartyProject(){                           //定义方法，配置 Smarty 模板
            $this->template_dir = "./";                   //指定模板文件存储在根目录下
            $this->compile_dir = "./system/Smarty/templates_c/";  //指定编译文件存储位置
            $this->config_dir = "./system/Smarty/configs/";
            $this->cache_dir = "./system/Smarty/cache/";
      }
}
?>
```

（2）既然已经将 Smarty 的配置方法存储到一个类中，那么就需要对类进行实例化，根据返回的对象名称调用 Smarty 中的方法，类的实例化操作在 system.inc.php 文件中完成，其返回对象名为$smarty。其代码如下：

```
<?php
      require("system.smarty.inc.php");                   //调用类文件
      $smarty=new SmartyProject();                        //执行类的实例化操作
?>
```

专家点评

通过此方法配置 Smarty 模板的好处是：无论将程序复制到哪个服务器下执行，都不需要更改服务器或 Smarty 文件的绝对路径，程序都可以直接运行。

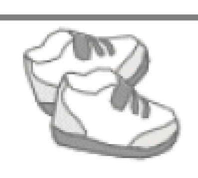
问题 250　如何通过 if 语句判断当前用户的权限？

问题阐述

在动态 PHP 文件中可以通过 SESSION 变量的值判断当前用户是否具有访问权限，而在 Smarty 模板页中不可以使用 SESSION 变量，那么该如何判断用户是否具有访问权限呢？

专家解答

在 Smarty 模板页中要判断用户权限可以使用 if 语句来实现，而关键是 if 语句和模板变量的结合运用，在 if 语句中判断模板变量的值是否为 T，如果为 T 则说明具备访问权限；如果模板变量值为 F 则说明不具备访问权限。





Smarty 模板中的 if 与 PHP 中的 if 大同小异。需要注意的一点是，Smarty 模板中的 if 必须以/if 为结束标记。其语法格式如下：

```
{if 条件语句 1}
    语句 1
{elseif 条件语句 2}
    语句 2
{else}
    语句 3
{/if}
```

应用示例

应用 if 语句在 Smarty 模板页中判断用户是否具有访问权限。其具体步骤如下：

（1）创建 system 文件夹，封装 Smarty 模板的配置方法，创建存储编译文件、缓存文件和配置文件的文件夹。

（2）创建 index.php 文件，包含 Smarty 配置文件，指定 Smarty 的模板页 index.html。

（3）创建 Smarty 模板页 index.html，设计用户登录页面，添加用户登录的表单元素，并且将用户的登录信息提交到 index_ok.php 文件。

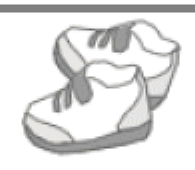
（4）创建 index_ok.php 文件，获取表单中提交的数据，判断提交的用户名和密码是否正确，如果正确则通过 display()方法指定模板页 main.html，并且为模板变量 competence 赋值为 T；否则指定到模板页 main.html，为模板变量 competence 赋值为 F。其关键代码如下：

```
<?php
header ( "Content-type: text/html; charset=UTF-8" );                    //设置文件编码格式
include("system/system.inc.php");                                       //包含配置文件
if($_POST['Submit']!="" && $_POST['user']!="" && $_POST['pass']!=""){
        if($_POST['user']=="mr" && $_POST['pass']=="mrsoft"){
            $smarty->assign("competence","T");                          //为模板变量赋值
            $smarty->display('main.html');                              //指定模板页
        }else{
            $smarty->assign("competence","F");
            $smarty->display('main.html');
        }
}else{
        echo "<script>alert('用户名和密码不能为空！'); window.location.href='index.php';</script>";
}
?>
```

（5）创建 main.html 模板页。首先应用 if 语句对模板变量的值进行判断，如果值为 T 则输出本页内容；如果值为 F 则给出提示信息。其关键代码如下：

```
{if $competence=="T"}
//省略了部分内容
{/if}
{if $competence=="F"}
<script>alert('您没有权限访问，请重新登录');window.location.href="index.php";</script>
{/if}
```

运行结果如图 14.3 所示。



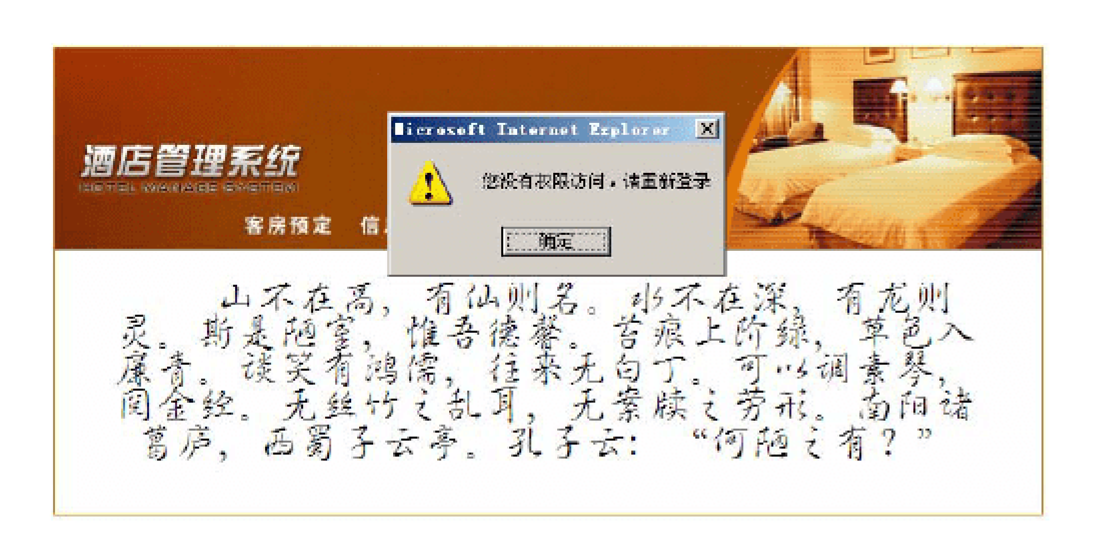


图 14.3 if 语句判断用户是否具备访问权限



Note

专家点评

Smarty 模板 if 条件语句的修饰词，除了可以使用 PHP 中的<、>、=、!=等常见运算符外，还可以使用 eq、ne、neq、gt、lt、lte、le、gte、ge、is even、is odd、is not even、is not odd、not、mod、div by、even by、odd by 等修饰词修饰。

问题 251 在 Smarty 模板中，如何生成数字验证码?

问题阐述

在动态 PHP 中可以生成数字验证码，在 Smarty 模板中，如何生成数字验证码?

专家解答

在 Smarty 模板中可以应用 foreach 语句在模板页中直接生成一个数字验证码。严格来说这个验证码不是在 Smarty 模板页中生成的，只是通过 foreach 语句将 PHP 动态页中生成的验证码在模板页中输出。因此要实现在 Smarty 中生成验证码的关键是 foreach 语句的应用和验证码的生成。

（1）应用 mt_rand()函数生成验证码。mt_rand()函数的语法如下：

```
int mt_rand ( [int min, int max])
```

如果 mt_rand()函数没有提供可选参数 min 和 max，则返回 0 到 RAND_MAX 之间的伪随机数。

（2）应用 explode()函数将生成的验证码写入数组中。explode()函数的语法如下：

```
array explode(string separator, string string, [int limit])
```

返回由字符串组成的数组，每个元素都是 string 的一个子串，它们被字符串 separator 作为边界点分隔出来。如果设置 limit 参数，则返回的数组包含最多 limit 个元素，而最后那个元素将包含 string 的剩余部分。

如果 separator 为空字符串（""），explode()函数将返回 false。

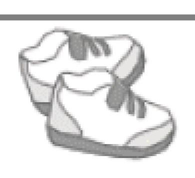
如果 separator 所包含的值在 string 中找不到，那么 explode()函数将返回包含 string 单个元素的数组。

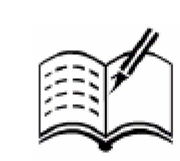
如果参数 limit 是负数，则返回除了最后的-limit 个元素外的所有元素。

（3）通过 assign()方法将验证码数组赋给模板变量。

（4）应用 foreach 语句循环输出模板变量中的数组元素。foreach 语法如下：

```
{foreach name=foreach_name key=key item=item from=arr_name}
…
{/foreach}
```





Note

参数说明：

❶ name：该循环的名称。

❷ key：当前元素的键值。

❸ item：当前元素的变量名。

❹ from：该循环的数组。

其中，item 和 from 是必要参数，不可省略。

应用示例

应用 Smarty 模板中的 foreach 语句在模板页中直接生成一个数字验证码，它相对于在动态 PHP 中开发的验证码更加简单、实用。其具体实现步骤如下：

（1）创建 system 文件夹，封装 Smarty 模板的配置方法，这里不再赘述。

（2）创建 index.php 文件，生成验证码，并且将验证码赋给模板变量，最后指定模板页。其代码如下：

```
<?php
require_once("system/system.inc.php");                    //包含配置文件
$array= explode(' ', mt_rand(1000,9999));                 //生成随机验证码
$smarty->assign('title','Smarty 模板中生成数字验证码');   //将指定数据赋给模板变量
$smarty->assign('content',$array);                        //将数组赋给模板变量
$smarty->display('index.html');                           //指定模板页
?>
```

（3）创建 index.html 模板文件，设计用户登录页面，添加用户登录的表单元素。同时应用 foreach 语句输出模板变量中传递的验证码。其关键代码如下：

```
<form id="form1" name="form1" method="post" action="index_ok.php">
        <input name="user" type="text" id="user" size="10" />
        <input type="hidden" name="checks" value="{foreach key=key item=item from=$content}
{$item}{/foreach}" />
      <input name="check" type="text" size="8" />
      <input name="pass" type="password" id="pass" size="10" />
      <input type="image" name="imageField" src="images/Blog_03_03.jpg" />
        <input type="image" name="imageField2" onclick="form.reset();return false;" src=
"images/Blog_03_06.jpg" />
</form>
```

Smarty 模板中生成的验证码运行结果如图 14.4 所示。

（4）创建 index_ok.php 文件，获取表单中提交的数据，对用户登录信息进行验证。如果用户名和密码正确，则跳转到 main.php 页面，否则跳转到 index.php 页面。

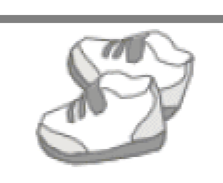
（5）创建 main.php 和 main.html 文件，并将其作为网站的主页。

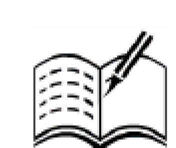


图 14.4　Smarty 模板中生成数字验证码

专家点评

本示例中应用 foreach 语句输出验证码，如果要更新验证码的值，则可以单击刷新按钮或





按 F5 键，同样也可以在地址栏中重新输入地址。

问题 252　在 Smarty 模板中，如何直接定义 CSS 样式？

问题阐述

由于在 Smarty 模板中默认使用“{”和“}”作为定界符，所以如果直接在 Smarty 模板中定义了 CSS 样式，那么将输出如图 14.5 所示的错误。

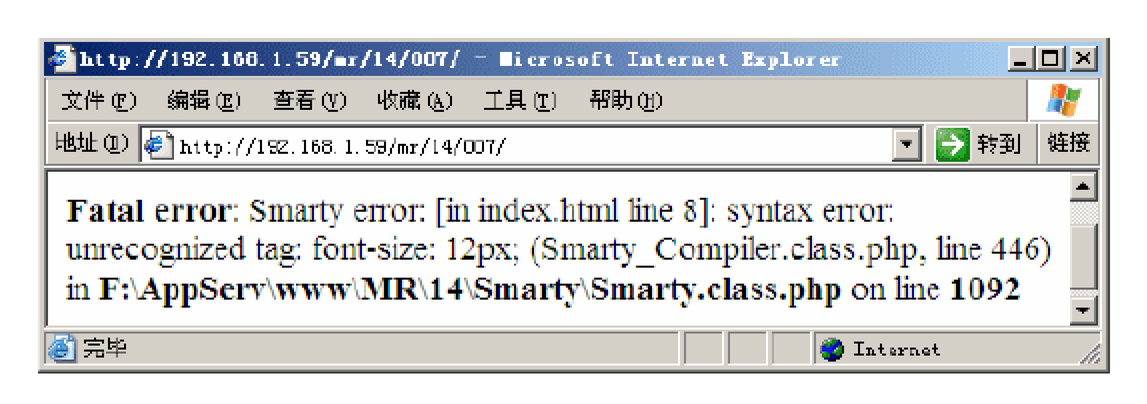


图 14.5　在 Smarty 模板中直接定义 CSS 样式时出现的错误

因为在 Smarty 模板页中直接将 CSS 样式中使用的“{”和“}”解析为 Smarty 模板的定界符，从而导致出现错误。那么如何避免在 Smarty 模板中直接定义 CSS 样式时出现错误呢？

专家解答

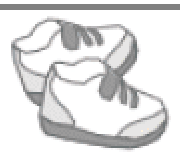
为了避免在 Smarty 模板中直接定义 CSS 样式时出现错误，Smarty 提供了特殊的处理方法，即 literal 标签，有了它就可以在 Smarty 模板页中直接定义 CSS 样式。

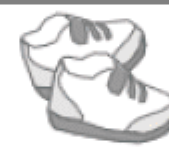
literal 标签将区域内的数据当作文本处理，此时模板将忽略其内部的所有字符信息。该特性用于显示有可能包含大括号等字符信息的 CSS 样式或 JavaScript 脚本，当这些信息处于{literal}{/literal}标签中时，模板引擎将不分析它们，而直接进行显示。

应用示例

这里仍然应用上一示例中的内容，唯一的区别是在 index.html 模板页中，通过{literal}{/literal}标签定义一个 CSS 样式，控制页面中表格、文字的背景、字体和大小等。其关键代码如下：

```
{literal}
<style type="text/css">
<!--
body,td,th {
	font-size: 12px;
}
body {
	margin-left: 0px;
	margin-top: 0px;
	margin-right: 0px;
	margin-bottom: 0px;
}
a:link {
	text-decoration: none;
}
a:visited {
	text-decoration: none;
}
a:hover {
```





Note

```
        text-decoration: none;
}
a:active {
        text-decoration: none;
}
-->
</style>
{/literal}
```

专家点评

在 Smarty 模板中，默认的定界符是“{”和“}”，为了避免与 CSS 样式或 JavaScript 脚本中的大括号冲突，还可以对 Smarty 模板的定界符进行修改，使用的是$left_delimiter 和$right_delimiter 结束符变量。其基本使用语法如下：

```
$smarty->left_delimiter = '<{';
$smarty->right_delimiter = '}>';
```

将定界符修改之后，就不会与 CSS 样式或 JavaScript 脚本中的大括号冲突了。

问题 253　在 Smarty 模板中，如何嵌入 JavaScript 脚本?

问题阐述

在 Smarty 模板中，如何嵌入 JavaScript 脚本呢？

专家解答

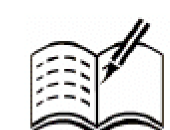
由于 Smarty 模板总是尝试对“{”和“}”进行解析，所以想要直接在 Smarty 模板中嵌入 JavaScript 脚本，就必须对“{”和“}”进行处理，前面已经介绍 literal 标签对大括号内的内容进行处理，在这里将介绍另外一种方法来输出 Smarty 模板中的定界符，进而实现在 Smarty 模板中直接嵌入 JavaScript 脚本，即应用 ldelim 和 rdelim 标签输出 Smarty 模板中的定界符，也就是说通过这两个标签在 Smarty 模板中直接输出“{”和“}”，从而完成在 Smarty 模板中直接嵌入 JavaScript 脚本的操作。

应用示例

在 index.html 模板页中直接定义 JavaScript 脚本，对表单中提交的数据进行判断，通过 ldelim 和 rdelim 标签在 Smarty 模板中输出“{”和“}”。其关键代码如下：

```
<script language=javascript>
function check_form() {ldelim}
        if (form.user.value == ''){ldelim}
                alert('请输入用户名');
            form.user.focus();
            return false;
        {rdelim}
        if (form.pass.value == ''){ldelim}
                alert('请输入密码');
            form.pass.focus();
```





```
                return false;
            {rdelim}
            if (form.check.value == ''){ldelim}
                alert('请输入验证码');
                form.check.focus();
                return false;
            {rdelim}
        {rdelim}
        </script>
```

专家点评

literal 标签与 delim、rdelim 标签的区别在于：literal 标签将“{”与“}”之间的内容按照普通的文本文件来解析，其控制的是内容；而 delim 与 rdelim 标签在 Smarty 模板中输出“{”与“}”，是对大括号本身进行操作。在 Smarty 模板中完成 CSS 样式或 JavaScript 脚本的嵌入操作时，笔者认为还是使用 literal 标签更方便一些。

问题 254　如何应用 html_option 函数向下拉列表中添加列表项？

问题阐述

在开发程序的过程中，都使用过下拉列表框，而且都知道下拉列表框的值是通过 option 标签定义的，可以直接在页面中定义，也可以使用从数据库中读取的数据来定义，无论使用哪种方法都必须应用 option 标签，这是在常规的 Web 页面中为下拉列表赋值的方法，那么在 Smarty 模板中如何为下拉列表框赋值呢？

专家解答

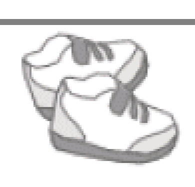
在 Smarty 模板中可以通过自定义函数 html_options 为下拉列表框赋值。自定义函数 html_options 根据给定的数据创建选项组。该函数可以指定哪些元素被选定，要么必须指定 values 和 ouput 属性，要么指定 options 替代。自定义函数语法如下：

```
<select name=customer_id>
    {html_options values=$cust_ids selected=$customer_id output=$cust_names}
</select>
```

html_options()函数的参数说明如表 14.1 所示。

表 14.1　html_options()函数的参数说明

参　　数	类　　型	说　　明
name	string	下拉菜单名称，默认值为空
options	associative array	包含值和显示的关联数组
selected	string/array	已选定的元素或元素数组
output	array	包含下拉列表各元素显示值的数组
values	array	包含下拉列表各元素值的数组



如果给定值是数组，则将作为 OPTGROUP 处理，且支持递归。所有的输出与 XHTML 兼容。

如果指定可选属性 name，则该选项列表将被置于<select name="groupname"></select>标签中。如果没有指定，那么只产生选项列表。



Note

应用示例

通过 Smarty 模板中的自定义函数 html_options 为下拉列表框赋值，更加适合从数据库中读取数据，更符合 Smarty 开发模式动静分离的原则。其具体实现步骤如下：

（1）创建 system 文件夹。首先，定义 system.smarty.inc.php 文件，封装 Smarty 的配置方法以及 ADODB 连接和操作数据库的方法。然后，定义 system.inc.php 文件，对 Smarty 配置类、数据库的连接和操作类进行实例化，并返回连接对象。最后，创建 Smarty 文件夹，定义 Smarty 的编译文件、配置文件和缓存文件的存储目录。

（2）创建 index.php 文件。首先，设置页面的编码格式、载入配置文件。然后，生成随机验证码，并且将随机验证码的值赋给模板变量。接着，从数据库中读取数据，将读取的数据存储到数组中，并且将数组赋给模板变量，作为下拉列表中的值。最后，指定模板页。其代码如下：

```
<?php
header ( "Content-type: text/html; charset=UTF-8" );                    //设置文件编码格式
require_once("system/system.inc.php");                                  //包含配置文件
$array= explode(' ', mt_rand(1000,9999));                               //生成随机验证码
$smarty->assign('title','html_option 函数向下拉列表中添加列表项');      //将指定数据赋给模板变量
$smarty->assign('content',$array);                                      //将数组赋给模板变量
$res=$conn->execute("select * from tb_bccd ");                          //执行 select 查询语句
$array=$res->GetArray();                                                //返回查询结果
$array_id=array();
$array_name=array();
for($i=0;$i<$res->RecordCount();$i++){
        $array_id[]=$array[$i][id];
        $array_name[]=$array[$i][name];
}
$smarty->assign('cust_id', $array_id);                                  //设置下拉列表的值
$smarty->assign('cust_name', $array_name);                              //设置列表的显示数据
$smarty->display('index.html');                                         //指定模板页
?>
```

（3）创建 index.html 模板页。首先，创建 form 表单，添加表单元素，将用户的注册信息提交到 index_ok.php 文件中，其中，通过 html_options 自定义函数输出下拉列表框的值，通过 foreach 语句输出验证码的值。然后，应用 literal 标签在页面中直接嵌入 CSS 样式和 JavaScript 脚本对用户提交的信息进行验证。其关键代码如下：

```
<form id="form" name="form" method="post" action="index_ok.php" onsubmit="return check_form(); ">
词典选择：<select name="customer" id="customer">{html_options output=$cust_name values= $cust_id }
</select>
验证码：<input type="hidden" id="checks" name="checks" value="{foreach key=key item=item
from=$content} {$item} {/foreach}" />
<input name="check" type="text" id="check" size="8" />{foreach key=key item=item from=$content}
{$item} {/foreach}
<input type="image" name="imageField3" src="images/reg_07.jpg" />
```



```
        <input  type="image"  name="imageField4"  onclick="form.reset();return  false;"  src="images/reg_
09.jpg" />
        </form>
```

生成验证码和下拉列表框的运行结果如图 14.6 所示。



图 14.6　通过 html_options()方法为下拉列表赋值

（4）创建 index_ok.php 文件，获取表单中提交的数据，并将数据添加到数据表中。其代码如下：

```
        <?php
        header ( "Content-type: text/html; charset=UTF-8" );                    //设置文件编码格式
        require_once("system/system.inc.php");                                  //包含配置文件
        if($_POST['user']!=""&&$_POST['pass']!=""&&$_POST['checks']!=""){//判断用户名和密码是否为空
                $sql="insert into tb_user(user,pass,dates,email,type)values('".$_POST['user']."','".md5($_POST
['pass'])."', '".date("Y-m-d")."','".$_POST['email']."','".$_POST['customer']."')";
                $res=$admindb->ExecSQL($sql,$conn);                             //执行 insert 添加语句
                if($res){
                    echo "<script>alert('用户注册成功！'); window.location.href='main.php';</script>";
                }else{
                    echo "<script>alert('用户注册失败！'); window.location.href='index.php';</script>";
                }
        }else{
                echo "<script>alert('用户注册信息不可为空！'); window.location.href='index.php';</script>";
        }
        ?>
```

（5）创建 main.php 和 main.html 文件，定义用户注册成功后跳转的页面。

专家点评

在 PHP 脚本中，如果没有对文件的编码格式进行设置，那么在输出 JavaScript 脚本时，如果输出的是中文字符串，那么将会输出乱码，如图 14.7 所示。

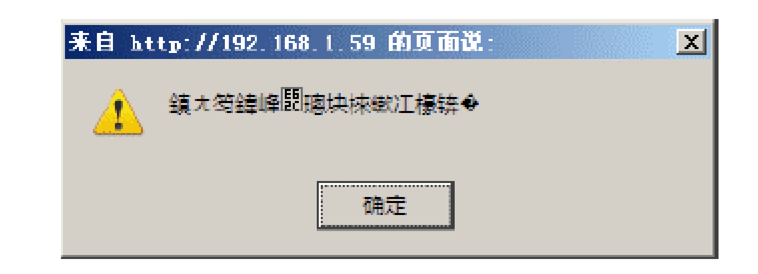
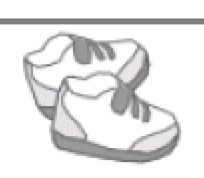


图 14.7　在 JavaScript 脚本中输出乱码

所以在 PHP 中通过 header()函数设置页面的编码格式还是十分必要的。其基本的使用方法如下：

```
        header ( "Content-type: text/html; charset=UTF-8" );              //设置文件编码格式
```



问题 255　section 语句应用的注意事项有哪些？



Note

问题阐述

section 是 Smarty 模板中的数组遍历语句，通过它能够完成对模板变量中传递的二维数组数据的循环输出。那么这个 section 语句是如何应用的呢？在应用过程中需要注意哪些问题呢？

专家解答

section 用于遍历数组中的数据。section 标签必须成对出现，且必须设置 name 和 loop 属性。名称可以是包含字母、数字和下划线的任意组合。

可以嵌套使用，但必须保证嵌套的 name 唯一。变量 loop（通常是数组）决定循环执行的次数。当需要在 section 循环内输出变量时，必须在变量后加上中括号包含着的 name 变量。

输出 section 循环内的变量的格式是{$arr_name[sec_name].字段名称}。

问题 256　如何通过 section 循环输出数据？

问题阐述

在 PHP 动态页中可以通过 while、do…while、for 和 foreach 语句实现数据的循环输出，而在 Smarty 中它也有属于自己的循环输出语句 foreach 和 section，那么如何通过 section 循环输出数据呢？

专家解答

section 循环语句用于比较复杂的数组。section 的语法如下：

```
{section name="sec_name" loop=$arr_name start=num step=num}
```

参数说明：

❶ name：表示循环的名称。

❷ loop：表示循环的数组。

❸ start：表示循环的初始位置，如果 start=2，那么说明循环是从 loop 数组的第 2 个元素开始的。

❹ step：表示步长，如果 step=2，那么当循环一次后，数组的指针将向下移动两位，以此类推。

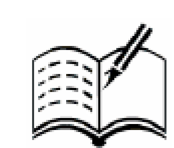
section 循环语句读取的是存储在模板变量中的数组元素，而这个数组元素值是在动态 PHP 文件中通过调用数据库操作类中的$admindb->ExecSQL 方法获取的。究其根源就是应用 ADODB 类库中的 GetRows()方法获取的查询结果。

应用示例

创建会员浏览模块（vip_look.php 和 vip_look.html）。应用 section 语句循环输出数据库中存储的会员信息。其具体步骤如下：

（1）编辑 vip_look.php 文件，定义 SQL 语句，调用数据库操作类（AdminDB）中的 ExecSQL





方法，查询出数据库中的会员数据，并且将返回的数组赋给指定的模板变量。其代码如下：

```
<?php
$sql="select * from tb_user ";                    //定义 SQL 查询语句
$res=$admindb->ExecSQL($sql,$conn);           //执行查询操作
if($res){
    $smarty->assign("res",$res);                  //将查询结果赋给指定的模板变量
}
?>
```

（2）编辑 vip_look.html 文件，应用 section 语句循环输出模板变量中传递的会员数据。其关键代码如下：

```
{section name=id loop=$res}
  <tr>
    <td style="padding-bottom:5px; padding-left:5px; padding-right:5px; padding-top:5px;" align="center" bgcolor="#FFFFFF">{$res[id].id}</td>
    <td style="padding-bottom:5px; padding-left:5px; padding-right:5px; padding-top:5px;" align="left" bgcolor="#FFFFFF">{$res[id].user}</td>
    <td style="padding-bottom:5px; padding-left:5px; padding-right:5px; padding-top:5px;" align="left" bgcolor="#FFFFFF">{$res[id].email}</td>
    <td style="padding-bottom:5px; padding-left:5px; padding-right:5px; padding-top:5px;" align="left" bgcolor="#FFFFFF">{$res[id].dates}</td>
  </tr>
{/section}
```

运行结果如图 14.8 所示。

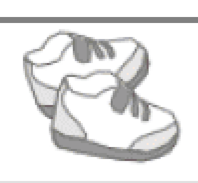


图 14.8　通过 section 循环输出数据

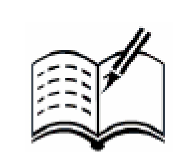
专家点评

在本示例中应用 section 语句循环输出数组中的所有元素值，同时还可以在 section 语句中添加第 3 个参数和第 4 个参数，即控制循环的开始位置和数组指针的移动位置。示例代码如下：

```
{section name=id loop=$res start=2 step=2}
  <tr>
    <td style="padding-bottom:5px; padding-left:5px; padding-right:5px; padding-top:5px;" align="center" bgcolor="#FFFFFF">{$res[id].id}</td>
    <td style="padding-bottom:5px; padding-left:5px; padding-right:5px; padding-top:5px;" align="left" bgcolor="#FFFFFF">{$res[id].user}</td>
    <td style="padding-bottom:5px; padding-left:5px; padding-right:5px; padding-top:5px;" align="left" bgcolor="#FFFFFF">{$res[id].email}</td>
    <td style="padding-bottom:5px; padding-left:5px; padding-right:5px; padding-top:5px;"
```



```
align="left" bgcolor="#FFFFFF">{$res[id].dates}</td>
            </tr>
          {/section}
```



Note

在上述的这个 section 语句中，将从数组的第 2 个元素开始循环，并且数组指针一次向下移动两位。还有就是在应用 section 语句时，切记一定要有{/section}结束标记。

问题 257　如何对 Smarty 模板中数据进行分页显示？

问题阐述

在 Smarty 模板中如何对数据进行分页显示？

专家解答

在 Smarty 中实现分页与在 PHP 文件中实现分页原理是相同的，唯一的区别是在 Smarty 中分页的处理操作与分页的显示是分离的。如分页的处理操作存储在 sho_delete.php 文件中。其应用到的关键技术如下：

（1）mysql_query()函数，执行查询、更改及删除等操作。语法如下：

```
resource mysql_query ( string query [, resource link_identifier] )
```

参数 query 为字符串类型，指定传入的 SQL 指令；参数 link_identfier 为资源类型，指定传入的是由 mysql_connect()函数或 mysq1_pconnect()函数返回的连接标识，如果省略该参数，则会使用最后一个打开的 MySQL 数据库连接。

（2）mysql_fetch_array()函数，从数组结果集中获取数据。语法如下：

```
array mysql_fetch_array ( resource result [, int result_type] )
```

参数说明：

❶ result：资源类型的参数，要传入的是由 mysql_query()函数返回的“数据指针”。

❷ result_type：可选项，整数型参数，要传入的是 MYSQL_ASSOC、MYSQL_NUM、MYSQL_BOTH 3 种由 PHP 定义好的常数之一，默认值是 MYSQL_BOTH。MYSQL_ASSOC 只得到关联索引（相当于 mysql_fetch_assoc()函数）；MYSQL_NUM 只得到数字索引（相当于 mysql_fetch_row()函数）；MYSQL_BOTH 将得到一个同时包含关联和数字索引的数组。

（3）array()函数；定义一个数组。返回根据参数建立的数组。参数可以用=>运算符给出索引。array()函数是一个语言结构，用于字面上表示数组，不是常规的函数。语法如下：

```
array array ( [mixed…])
```

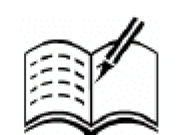
参数 mixed 的语法为“key => value”，多个参数 mixed 用逗号分开，分别定义了索引和值。

索引可以是字符串或数字。如果省略了索引，会自动产生从 0 开始的整数索引。如果索引是整数，则下一个产生的索引将是目前最大的整数索引+1。如果定义了两个完全一样的索引，则后面一个会覆盖前一个。

（4）array_push()函数，将数组当成一个栈，并将传入的变量压入该数组的末尾。该数组的长度将增加入栈变量的数目。返回数组新的单元总数。语法如下：

```
int array_push ( array array, mixed var [, mixed…])
```





参数说明：

❶ array：必要参数。输入的数组。

❷ var：必要参数。用来压入数组的值。

❸ var2：可选参数。用来压入数组的值。

（5）通过 PHP 中的函数、方法将从数据库中读取的数据进行分页处理，通过$smarty 对象调用 assign 方法将分页处理返回的变量传递给模板变量，并通过 display 方法指定模板页。

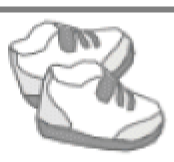
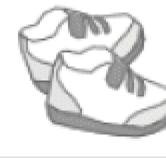
（6）在模板页 sho_delete.html 中获取模板变量传递的值，完成数据的分页输出。

应用示例

本示例在 section 语句循环输出数据库中数据的基础上增加一个分页功能，实现对数据库中数据的分页显示。其具体方法如下：

（1）编辑 sho_delete.php 文件，连接数据库、载入 Smarty 配置文件，通过 SQL 语句读取数据库中的数据，并且定义数据的分页输出方法，应用 Smarty 中的 assign 方法将分页输出的变量值传递给模板页，最终指定模板页。其代码如下：

```
<?php
include_once "conn/conn.php";                              //连接数据库
require_once("system/system.inc.php");                     //调用指定的配置文件
//执行查询语句，从数据库中读取商品信息
$sql=mysql_query("select count(*) as total1 from tb_bccd ",$conn);
$info=mysql_fetch_array($sql);
$total1=$info[total1];                                     //统计数据库中数据总数
if(empty($_GET[pages])==true || is_numeric($_GET[pages])==false){ //判断变量 pages 是否为空
    $page1=1;                          //如果变量为空，则赋值为 1
}else{                                 //如果不为空，则获取变量的值
    $page1=intval($_GET[pages]);
}
    $pagesize1=3;                      //定义每页显示 3 条记录
    if($total1<$pagesize1){            //判断如果数据库中数据小于每页显示的记录数
        $pagecount1=1;                 //则定义 pagecount 变量的值为 1
    }else{
        if($total1%$pagesize1==0){
            $pagecount1=intval($total1/$pagesize1);   //用总的记录数除以每页显示的记录数，
                                                       获取共有几页
        }else{
            $pagecount1=intval($total1/$pagesize1)+1;
        }
    }
//将要输出的数据赋给 assign 模板变量
$smarty->assign("total1",$total1);
$smarty->assign("pagesize1",$pagesize1);
$smarty->assign("page1",$page1);
$smarty->assign("pagecount1",$pagecount1);
$query=mysql_query("select * from tb_bccd order by id desc limit ".($page1-1)*$pagesize1.",
$pagesize1",$conn );
$myrow=mysql_fetch_array($query);
$array=array();                                 //定义一个空数组
do{
   array_push($array,$myrow);                   //将获取的数组值写入到新的数组中
}while($myrow=mysql_fetch_array($query));       //循环读取数据库中的数据
```

```
        if(!$array){
                $smarty->assign("iscommo","F");//判断如果执行失败则输出模板变量 iscommo 的值为 F
        }else{
                $smarty->assign("iscommo","T");//判断如果执行成功，则输出模板变量 iscommo 的值为 T
                $smarty->assign("arr",$array);    //定义模板变量 arr，输出数据库中的数据
        }
        $smarty->assign(title,"商品删除");
        ?>
```

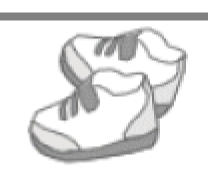
（2）编辑 sho_delete.html 模板页，应用 section 语句和模板变量完成数据的分页输出。其关键代码如下：

```
        <table width="545" border="1" cellpadding="1" cellspacing="1" bordercolor="#FFFFFF" bgcolor=
"#3399CC">
          <tr>
            <td height="25" align="center" bgcolor="#FFFFFF">ID</td>
            <td align="center" bgcolor="#FFFFFF">商品名称</td>
            <td align="center" bgcolor="#FFFFFF">操作</td>
          </tr>
          {section name=id loop=$arr}
          <tr>
            <td align="center" bgcolor="#FFFFFF">{$arr[id].id}</td>
            <td bgcolor="#FFFFFF">{$arr[id].name}</td>
            <td align="center" bgcolor="#FFFFFF"><a href="delete.php?conn_id={$arr[id].id}">删除</a></td>
          </tr>
          {/section}
        </table>
        <table width="545" height="25" border="0" align="center" cellpadding="0" cellspacing="0">
          <tr>
            <td>图书 {$total1} 本 每页{$pagesize1}本 第 {$page1} 
页/共 {$pagecount1} 页</td>
        <td align="center">{if $page1 == 1 }首页 上一页{else}<a href="index.php?caption=
{$title|escape:"url"} &pages=1">首页</a> 
        <a href="index.php?caption={$title|escape:"url"}&pages={$page1-1}" >上一页</a>{/if} {if $page1
== $pagecount1 }下一页 尾页{else}
        <a href="index.php?caption={$title|escape:"url"}&pages={$page1+1}">下一页</a> 
        <a href="index.php?caption={$title|escape:"url"}&pages={$pagecount1}">尾页</a>{/if}
        </td>
        </tr>
        </table>
```

运行结果如图 14.9 所示。



图 14.9　Smarty 模板中数据的分页输出





问题 258　在 Smarty 模板中，如何实现日期、时间的格式化输出？

问题阐述

在 PHP 脚本中日期、时间的格式化输出最常用的就是 date()函数，那么在 Smarty 模板中，如何完成日期、时间的输出呢？

专家解答

在 Smarty 模板中应用 date_format 函数，格式化 strftime()函数获得的时间和日期。其应用示例如下：

```
{$smarty.now|date_format:"%A, %B %e, %Y":$times}
```

参数说明：

❶ $smarty.now：传递给 date_format 的数据。这个参数可以是在 PHP 动态页中定义的日期、时间模板变量，也可以使用 Smarty 中的保留变量$smarty.now 或 Smarty 模板中的日期。

❷ %A, %B %e, %Y：date_format 函数执行格式化操作时使用的格式。

❸ $times：当传递给 date_format 的数据为空时，通过$times 设置 date_format 格式化的默认值。

date_format 函数可以使用的转换格式有很多，其常用的转换格式说明如下：

```
%a - 根据当地格式输出“星期”缩写格式
%A - 根据当地格式输出“星期”全称格式
%b - 根据当地格式输出“月”缩写格式
%B - 根据当地格式输出“月”全称格式
%Y - 根据当地格式输出“年”全称格式
```

要了解 date_format 函数所有可以使用的转换格式，建议读者参考 Smarty 模板的参考手册。

应用示例

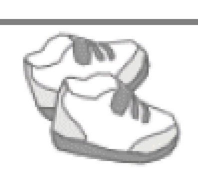
在 index.html 模板文件中插入 date_format 函数，输出系统的当前时间。其关键代码如下：

```
<td width="480" height="18" align="left">当前时间：{$smarty.now|date_format:" %A   %B -%e - %Y"}</td>
<!--{$smarty.now|date_format}
{$smarty.now|date_format:"%A, %B %e, %Y":$times}
{$smarty.now|date_format:"%H:%M:%S"}
{$yesterday|date_format}
{$yesterday|date_format:"%A, %B %e, %Y"}
-->
```

专家点评

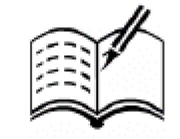
通过 date_format 函数不但可以输出系统的当前时间，如果配合 strtotime()函数，还可以获取其他的时间。例如，获取系统前一天的时间，其方法如下：

首先在 index.php 中，应用 strtotime()函数获取前一天的时间戳，并且将时间戳赋给模板变量，



然后在 index.html 模板页中，应用 date_format 函数格式化输出前一天的时间。其关键代码如下：

```
//Index.php 文件
$smarty->assign('yesterday', strtotime('-1 day'));
//Index.html 文件
{$ yesterday |date_format:" %A   %B - %e - %Y"}
```



问题 259　在 Smarty 模板中，如何应用正则表达式？

问题阐述

在 Smarty 模板中，如何应用正则表达式？

专家解答

在 Smarty 模板中通过 regex_replace 方法调用正则表达式完成字符串的匹配操作。regex_replace 可对指定的字符串进行匹配，其包括两个参数：第一个是指定的正则表达式；第二个是指定的替换文本。其应用示例如下：

```
{$name|regex_replace:"/[mr]/":"<font color='#FF0000'>明日科技</font>"}
```

在这个示例中，将模板变量$name 中的“mr”替换为“明日科技”。

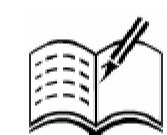
应用示例

在会员浏览模块中，应用 regex_replace 变量将会员名称中的“mr”替换为“明日科技”。其具体步骤如下：

（1）编辑 vip_look.php 文件，分页读取数据库中存储的会员数据，并且将分页读取返回的变量赋给模板变量，最终指定模板页。其关键代码如下：

```
<?php
include_once "conn/conn.php";                    //连接数据库
require_once("system/system.inc.php");           //调用指定的文件
//执行查询语句，从数据库中读取商品信息
$sql=mysql_query("select count(*) as total1 from tb_name ",$conn);
$info=mysql_fetch_array($sql);
$total1=$info[total1];                           //统计数据库中数据总数
if(empty($_GET[pages])==true || is_numeric($_GET[pages])==false){ //判断变量 pages 是否为空
        $page1=1;                                //如果变量为空，则赋值为 1
}else{                                           //如果不为空，则获取变量的值
        $page1=intval($_GET[pages]);
}
    $pagesize1=3;                                //定义每页显示 3 条记录
    if($total1<$pagesize1){                      //判断如果数据库中数据小于每页显示的记录数
        $pagecount1=1;                           //则定义 pagecount 变量的值为 1
    }else{
        if($total1%$pagesize1==0){
            $pagecount1=intval($total1/$pagesize1);    //用总的记录数除以每页显示的记录数，
                                                       获取共有几页
        }else{
            $pagecount1=intval($total1/$pagesize1)+1;
```






```
            }
        }
        //将要输出的数据赋给 assign 模板变量
        $smarty->assign("total1",$total1);
        $smarty->assign("pagesize1",$pagesize1);
        $smarty->assign("page1",$page1);
        $smarty->assign("pagecount1",$pagecount1);
        $query=mysql_query("select * from tb_name order by id desc limit ".($page1-1)*$pagesize1.",
$pagesize1",$conn );
        $myrow=mysql_fetch_array($query);
        $array=array();                                //定义一个空数组
        do{
            array_push($array,$myrow);                 //将获取的数组值写入新的数组中
        }while($myrow=mysql_fetch_array($query));      //循环读取数据库中的数据
        if(!$array){
                $smarty->assign("iscommo","F");//判断如果执行失败则输出模板变量 iscommo 的值为 F
        }else{
                $smarty->assign("iscommo","T");//判断如果执行成功，则输出模板变量 iscommo 的值为 T
                $smarty->assign("arr",$array);    //定义模板变量 arraybbstell，输出数据库中的数据
        }
        $smarty->assign(title,"会员浏览");
        ?>
```

（2）编辑 vip_look.html 模板页，通过 section 语句完成会员信息的分页输出，并且应用 regex_replace 变量将会员名称中的“mr”替换为“明日科技”。其关键代码如下：

```
        {section name=id loop=$arr}
          <tr>
            <td align="center" bgcolor="#FFFFFF">{$arr[id].id}</td>
            <td align="left" bgcolor="#FFFFFF">{$arr[id].name|regex_replace:"/[mr]/":"<font color='#FF0000'>
明日科技</font>"}</td>
            <td align="left" bgcolor="#FFFFFF">{$arr[id].email}</td>
            <td align="left" bgcolor="#FFFFFF">{$arr[id].dates}</td>
          </tr>
        {/section}
```

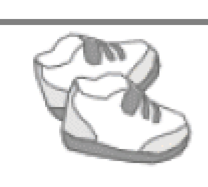
运行结果如图 14.10 所示。



图 14.10　Smarty 模板中正则表达式的应用

专家点评

在 Smarty 模板中不但通过 regex_replace 可以实现字符串的替换操作，而且使用 replace 函



数也可以实现字符串的替换操作。

问题 260　在 Smarty 模板中，如何对关键字描红？



问题阐述

在动态 PHP 中，可以通过函数 str_replace 实现对关键字描红，在 Smarty 模板中，如何对关键字描红呢？

专家解答

（1）Smarty 模板中的关键字描红应用的是 replace 变量，其实现字符串的简单查询和替换操作。其包括两个参数：第一个参数是被替换的文本字符串；第二个参数是用来替换的文本字符串。其应用示例如下：

```
{$articleTitle|replace:"Garden":"Vineyard"}
```

在这个示例中，将模板变量$articleTitle 中的 Garden 替换为 Vineyard。

（2）查询关键字描红的原理：在 PHP 动态页中获取 form 表单提交的查询关键字，并且执行模糊查询，将查询结果赋给模板变量，同时将查询的关键字也赋给模板变量，然后在模板页中通过 section 语句循环输出查询结果，并应用 replace 将查询结果中关键字的字体设置为红色。

应用示例

在后台管理的订单查询模块中对查询结果中的关键字进行描红。其具体步骤如下：

（1）编辑 for_select.html 模板页，创建 form 表单，提交查询的关键字，将数据提交到 for_select.php 页面进行处理；同时在本页中应用 section 语句循环输出查询结果，并且应用 replace 对查询的关键字进行描红操作。其关键代码如下：

```
{section name=id loop=$res}
<tr>
    <td align="center" bgcolor="#FFFFFF">{$res[id].id}</td>
    <td bgcolor="#FFFFFF">{$res[id].number|replace:$select:"<font color='#FF0000'>$select</font>"}
</td>
    <td bgcolor="#FFFFFF">{$res[id].user|replace:"明日科技":"<font color='#FF0000'>明日科技
</font>"}</td>
    <td bgcolor="#FFFFFF">{$res[id].dates}</td>
    <td align="center" bgcolor="#FFFFFF">{if $res[id].type==0}未处理{else}已处理{/if}</td>
</tr>
{/section}
```

（2）编辑 for_select.php 动态页，以 form 表单中提交的关键字为条件执行模糊查询，如果查询结果为真，则将查询结果赋给模板变量，同时将查询的关键字也赋给指定的模板变量；如果查询结果为假，则将为模板变量赋值 F，其代码如下：

```
<?php
$smarty->assign(title,"Smarty 模板中的关键字描红技术");       //定义标题变量
if($_POST['number']!=""){                                      //判断查询关键字是否为空
    $sql="select * from tb_order where number like '%".$_POST['number']."%'"; //定义模糊查询的语句
    $res=$admindb->ExecSQL($sql,$conn);                        //执行模糊查询
```



```
        if($res){
            $smarty->assign("select",$_POST['number']);            //将查询的关键字赋给模板变量
            $smarty->assign("boo","T");                           //将模板变量赋值为 T
            $smarty->assign("res",$res);                          //将查询结果赋给模板变量
        }else{
            $smarty->assign("boo","F");
        }
    }else{
        $smarty->assign("boo","F");
    }
    ?>
```

运行结果如图 14.11 所示。



图 14.11　Smarty 模板中的关键字描红

专家点评

通过 replace 变量不但可以实现查询关键字描红的功能，而且可以实现对论坛中的非法关键字进行替换的功能，其实现方法与关键字的描红相同。

问题 261　在 Smarty 模板中，如何控制输出字符串的行宽？

问题阐述

在 Web 页面中通常是应用 CSS 样式来控制字符串输出的行宽，那么在 Smarty 模板中，如何控制输出字符串的行宽呢？

专家解答

在 Smarty 模板中有一个变量可以控制输出字符串的行宽，它就是 wordwrap。

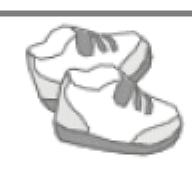
wordwrap 指定段落的宽度（即控制一行显示多少个字符，如果超过这个字符数则换行），其默认值是 80 字符。该方法包含 3 个参数：

❶ 第一个参数设置每行显示的字符数。

❷ 第二个参数设置在约束点使用什么字符（默认值是换行符\n）。

❸ 第三个参数控制 Smarty 是截取到字符串的末尾，还是精确到指定长度的字符，默认情况是截取到字符串的默认值。如果设置该参数值为 true，那么就精确截取到指定长度的字符。其应用的示例如下：

```
{$articleTitle|wordwrap:30}
```

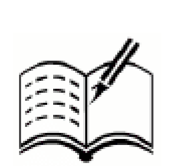


设置段落一行中包括 30 个字符，并使用\n 进行换行操作。

```
{$articleTitle|wordwrap:30:"<br>\n"}
```

设置段落一行中包括 30 个字符，并使用
\n 进行换行操作。

```
{$articleTitle|wordwrap:30:"\n":true}
```



Note

设置段落一行中包括 30 个字符，并使用\n 进行换行操作，同时精确截取 30 个字符。

上述的示例是对模板变量$articleTitle 中存储的数据进行操作。

应用示例

在本示例中应用 wordwrap 变量控制页面中输出字符串的行宽。其具体步骤如下：

（1）创建 system 文件夹。首先，定义 system.smarty.inc.php 文件，封装 Smarty 的配置方法。然后，定义 system.inc.php 文件，对 Smarty 配置类进行实例化，并返回连接对象。最后，创建 Smarty 文件夹，定义 Smarty 的编译文件、配置文件和缓存文件的存储目录。

（2）创建 index.php 文件。首先，设置页面的编码格式、载入配置文件。然后，通过 file_get_contents()函数读取存储在文本文件 content.txt 中的字符串，并应用 iconv()函数将字符串有 gb2312 编码转换为 UTF-8 编码，同时将转换后的文件存储到模板变量中，最终指定模板页。其代码如下：

```
<?php
header ( "Content-type: text/html; charset=UTF-8" );                //设置文件编码格式
include("system/system.inc.php");                                   //包含配置文件
$smarty->assign('title','Smarty 模板中控制输出字符串的行宽');        //定义实例标题变量
$str = file_get_contents('files/content.txt');                      //读取文本文件中的数据
$smarty->assign('content',iconv("gb2312","utf-8",$str));            //将文本文件中的数据存储到模板变量中
$smarty->display('index.html');                                     //指定模板页
?>
```

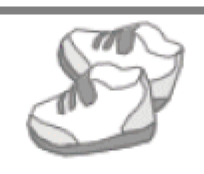
（3）创建 index.html 模板页，通过模板变量输出文本文件中的字符串，并且应用 wordwrap 控制每行显示 30 个字符，应用
\n 完成换行操作。其关键代码如下：

```
{$content|wordwrap:30:"<br>\n"}
```

运行结果如图 14.12 所示。



图 14.12 每行显示 30 个字符串





专家点评

在应用 wordwrap 控制段落中每行显示的字符时，默认使用\n 为换行符，但当使用\n 作为换行符时，对页面中字符串的输出不会受到影响，即看不到字符串换行的效果，只有在查看源文件时才能看出字符串已经换行，所以，在应用 wordwrap 控制段落中每行显示的字符时，应该设置第二个参数，指定
为换行符，这样在页面中输出时就可以直接看到段落换行的效果。

另外，在 Smarty 模板中还可以通过 nl2br 将所有的换行符转换成</br>，其应用与 PHP 中的 nl2br()函数相同。而本实例中的字符串换行操作也可以使用如下方法来完成：

```
{$content|wordwrap:30|nl2br}
```

问题 262　如何应用 register_object 方法注册模板对象？

问题阐述

在 PHP 中，可以通过 function 创建自定义函数，完成一些特殊的操作，那么在 Smarty 中如何通过 register_object 方法注册对象，使之在模板页中完成一些特殊的操作呢？

专家解答

使用 register_object 方法注册一个在模板中使用的对象，语法如下：

```
void register_object (string object_name, object $object, array allowed methods/properties, boolean format, array block methods)
```

register_object 注册对象的参数说明如表 14.2 所示。

表 14.2　register_object 注册对象的参数说明

参　　数	说　　明
object_name	模板页中使用的对象名称
$object	实例化的对象名称
allowed methods/properties	定义允许使用的方法，如果设置为null，表示所有方法都可以
format	设置对象参数传递的方法，如果值为false，则表示一次传递一个参数
block methods	该参数在format的值为true应用，向对象中添加一个块方法

应用示例

在本示例中通过 register_object 方法注册对象，定义两个方法 moneyFormat()和 unHtml()，分别对留言本中的评分进行转义，保留两位小数；对留言内容中的 html 标记进行转义。其具体步骤如下：

（1）创建 system 文件夹，定义 Smarty 文件夹，存储编译目录、缓存目录；创建类文件 system.smarty.inc.php，定义数据库连接、操作、分页和 Smarty 的配置类，定义类的实例化文件 system.inc.php，完成各个类的实例化操作，并返回操作对象。

（2）创建 index.php 文件，包含类的实例化操作文件；创建类，定义评分转义和 html 标记转义的方法，通过 register_object 注册模板对象；完成数据库中数据的分页读取，最终指定模板页。其代码如下：





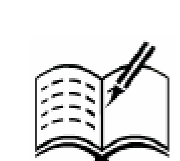

```
<?php
require_once("system/system.inc.php");            //调用指定的文件
class Util {
        function moneyFormat($integral) {          //将数字转换成有两位小数的形式
            return str_replace ( ',', '', number_format ( $integral, 2 ) );
        }
        function unHtml($text) {                   //转换字符串中的 html 标签
            $str = htmlspecialchars ( $text );
            $str = ereg_replace ( '<br>', '\n', $str );
            $str = nl2br ( $str );
            $str = ereg_replace ( ' ', ' ', $str );
            return $str;
        }
}
$ntil=new Util();                                  //实例化对象
$smarty->register_object("Util", $ntil,null,false);   //注册模板对象
$array=$seppage->ShowDate("select*from tb_guestbook order by id desc",$conn,2,$_GET["page"]);
                                                   //调用分页类，实现分页功能
if(!$array){
        $smarty->assign("iscommo","F");//判断如果执行失败，则输出模板变量 iscommo 的值为 F
}else{
        $smarty->assign("iscommo","T");//判断如果执行成功，则输出模板变量 iscommo 的值为 T
        $smarty->assign("showpage",$seppage->ShowPage("留言","条","","a1"));
                                                   //定义输出分页数据的模板变量 showpage
    $smarty->assign("array",$array);
}
    $smarty->assign('title','register_object()方法注册模板对象');
    $smarty->display('index.html');
?>
```

（3）创建 index.html 模板页，分页输出模板变量中传递的数据，并且调用注册模板对象中的方法对评分和 HTML 标记进行转换操作。同时创建一个 form 表单，用于提交留言信息，将数据提交到 index_ok.php 文件中。其关键代码可参考关键技术中的内容。

（4）创建 index_ok.php 文件，将表单中提交的留言信息存储到指定的数据表中。其代码如下：

```
<?php
header ( "Content-type: text/html; charset=UTF-8" );     //设置文件编码格式
require_once("system/system.inc.php");                   //调用指定的文件
$title=$_POST[title];                                    //获取标题
$content=$_POST[content];                                //获取内容
$integral=$_POST[integral];
$createtime=date("Y-m-j H:i:s");                         //定义时间
$arr=$admindb->ExecSQL("insert into tb_guestbook(title,content,createtime,integral) values
('$title','$content','$createtime','$integral')",$conn);
if($arr){
    echo "<script>alert('留言发表成功!');window.location.href='index.php';</script>";
}else{
     echo "<script>alert('留言发表失败!');history.back();</script>";
}
?>
```





运行结果如图 14.13 所示。



图 14.13 Smarty 模板中日期的格式化输出

专家点评

虽然 register_object 方法注册的对象在模板页中完成对数据的操作，但是其具体功能的实现仍然是在 PHP 动态文件中。下面应用 str_replace()和 number_format()函数实现字符串的替换和数字的格式化。

（1）str_replace()函数，实现字符串的替换。语法如下：

```
mixed str_replace ( mixed search, mixed replace, mixed subject , int &count )
```

str_replace()函数将所有在参数 subject 中出现的 search 以参数 replace 替换，参数&count 表示替换字符串执行的次数。str_replace()函数的参数说明如表 14.3 所示。

表 14.3 str_replace()函数的参数说明

参　　数	说　　明
search	指定将要被替换的字符
replace	指定替换所使用的字符
subject	指定被操作的字符串
&count	替换字符串执行的次数

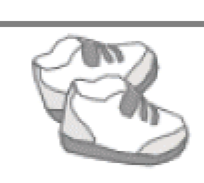
（2）number_format()函数，将数字字符串格式化。语法如下：

```
string number_format ( float number , int decimals)
string number_format ( float number, int decimals, string dec_point, string thousands_sep)
```

number_format()函数返回参数 number 格式化后的字符串，该函数可以有 1 个、2 个或 4 个参数，但不能是 3 个参数。

如果只有一个参数 number，则 number 格式化后会舍去小数点后的值，且每一千就会以逗号（,）来隔开。

如果有两个参数，则 number 格式化后会到小数点第 decimals 位，且每一千就会以逗号（,）来隔开。



如果有 4 个参数，则 number 格式化后会到小数点第 decimals 位，dec_point 用来替代小数点（.），thousands_sep 用来替代每一千隔开的逗号（,）。



问题 263　如何开启网站页面的缓存?

问题阐述

缓存的合理应用应该以实际的开发需要为依据，对于那些不需要经常更新的程序开启缓存，可以提高网站的访问效率，那么在 Smarty 模板中是如何开启网站缓存的呢？

专家解答

缓存的设置既要考虑网站访问速度的提高，又要考虑缓存生命周期的合理性。不能偏执一方，应该采取中庸之道，达到一个最理想的效果。

例如，网站首页是不需要经常更新的内容，缓存的生存周期就可以设置的长一些，而类似于论坛中帖子的数据，则可以不开启缓存，因为帖子的数据会不断地更新，就没有必要使用缓存，如果使用缓存可能会导致浏览不到最新的数据。

1. 开启缓存

开启缓存的方法非常简单，只要将 Smarty 对象中$config 的值设置为 True 即可，同时还要通过 Smarty 对象中的$cache_dir 属性指定缓存文件的存储位置。其操作代码如下：

```
$smarty->caching=true;                                       //开启缓存
$smarty->cache_dir = BASE_PATH.SMARTY_PATH.'cache/';   //定义缓存文件存储位置
```

2. 设置缓存的生命周期

缓存创建成功后，还必须为它设置一个生命周期，如果它一直不更新，那么就没有任何意义。设置缓存生命周期应用的是 Smarty 对象中的$cache_lifetime 属性，缓存时间以秒为单位，默认值是 3600 秒。其操作代码如下：

```
$smarty->caching=true;                                       //开启缓存
$smarty->cache_dir = BASE_PATH.SMARTY_PATH.'cache/';   //定义缓存文件存储位置
$smarty->cache_lifetime=3600                                 //设置缓存时间为 1 小时
```

3. 判断模板文件是否已经被缓存

如果页面已经被缓存，那么就可以直接调用缓存文件，而不再执行动态获取数据和输出的操作。为了避免在开启缓存后再次执行动态获取数据和输出操作给服务器带来的压力，最佳的方法就是应用 Smarty 对象中的 is_cached()方法判断指定的模板是否存在缓存，如果存在则直接执行缓存中的文件，否则执行动态获取数据和输出的操作。操作代码如下：

```
$smarty->caching=true;                                       //开启缓存
if(!$smarty->is_cached('index.html')){
        //执行动态获取数据和输出的操作
}
$smarty->display('index.html');
```

4. 清除模板中的缓存

缓存的清除有两种方法：



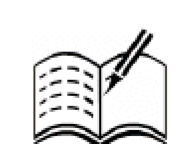
第一种是 clear_all_cache()方法，清除所有模板缓存。其语法如下：

```
void clear_all_cache (int expire time)
```

可选参数 expire time，可以指定一个以秒为单位的最小时间，超过这个时间的缓存都将被清除。

第二种是 clear_cache()方法，清除指定模板的缓存。

```
void clear_cache (string template [, string cache id [, string compile id [, int expire time]]])
```



如果这个模板有多个缓存，则既可以用第二个参数指定要清除缓存的缓存号，也可以通过第三个参数指定编译号。可以把模板分组，以便可以方便地清除一组缓存。第四个参数是可选的，用来指定超过某一时间（以秒为单位）的缓存才会被清除。

5. 应用示例

在本示例中开启网站注册页面的缓存。具体操作在 index.php 文件中完成，其代码如下：

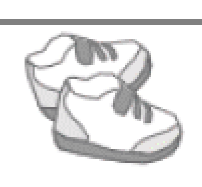
```
<?php
header ( "Content-type: text/html; charset=UTF-8" );          //设置文件编码格式
require_once("system/system.inc.php");                        //包含配置文件
$smarty->caching=true;                                        //开启缓存
$smarty->cache_lifetime=3600;                                 //设置缓存过期时间
//$smarty->clear_all_cache();                                 //清除所有缓存
if(!$smarty->is_cached('index.html')){
$res=$conn->execute("select * from tb_bccd ");                //执行 select 查询语句
$array=$res->GetArray();
$array_id=array();
$array_name=array();
for($i=0;$i<$res->RecordCount();$i++){
    $array_id[]=$array[$i][id];
    $array_name[]=$array[$i][name];
}
$smarty->assign('cust_id', $array_id);                        //设置下拉列表的值
$smarty->assign('cust_name', $array_name);                    //设置列表的显示数据
}
$array= explode(' ', mt_rand(1000,9999));                     //生成随机验证码
$smarty->assign('title','开启网站注册页面的缓存');            //将指定数据赋给模板变量
$smarty->assign('content',$array);                            //将数组赋给模板变量
$smarty->assign('contents',$arrays);                          //将数组赋给模板变量
$smarty->display('index.html');                               //指定模板页
?>
```

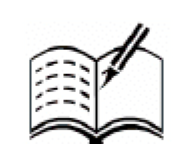
缓存文件存储在 system\Smarty\cache 目录下，而存储路径的设置是在 system\system.smarty.inc.php 文件中完成的。

问题 264　如何判断同一模板中的多个缓存文件？

问题阐述

在实际的程序开发中经常会遇到这样的情况：同一个模板文件生成多个页面。那么就要生成多个缓存文件，这多个缓存文件是如何生成的呢？又如何判断同一模板中的多个缓存文件呢？





专家解答

多个页面进行缓存应用的是 Smarty 中的 display()方法，通过该方法的第二个参数设置缓存号，有几个不同的缓存号就有几个缓存页面。操作代码如下：

```
$smarty->display('index.html',$_GET['id']);                //将 id 作为第二个参数传递
```

判断同一模板中的多个缓存是否存在，其依据的是缓存号。判断同一模板的多个缓存是否存在可应用 is_cached()方法，通过该方法的第二个参数设置缓存号，判断对应的缓存是否存在。其方法如下：

```
$smarty->caching=true;                                     //开启缓存
$smarty->cache_dir = BASE_PATH.SMARTY_PATH.'cache/';       //定义缓存文件存储位置
$smarty->cache_lifetime=3600;                              //设置缓存时间为 1 小时
if(!$smarty->is_cached('index.html',$_GET['id'])){
    //执行动态获取数据和输出的操作
}
$smarty->display('index.html',$_GET['id']);                //将 id 作为第二个参数传递
```

问题 265　如何通过配置文件定义变量？

问题阐述

配置文件的应用有利于设计者管理文件中的模板全局变量。例如，定义一个模板色彩变量。一般情况下如果想改变一个程序的外观色彩，则必须更改每一个文件的颜色变量。如果有配置文件，色彩变量就可以保存在一个单独的文件中，只要改变配置文件就可以实现色彩的更新，这与在 Web 页面开发中应用的 CSS 样式非常相似，那么如何通过配置文件定义变量呢？

专家解答

通过配置文件定义变量主要有以下 3 个步骤。

1. 创建配置文件

配置文件可以任意命名，其存储位置由 Smarty 对象的$config_dir 属性指定。如果存在不只在一个区域内使用的变量值，可以使用三引号（"""）将它完整地封装起来。在创建配置文件时，建议在程序运行前使用“#”加一些注释信息，这样有助于程序的阅读、更新。

在配置文件中既可以声明全局变量，也可以声明局部变量。如果声明局部变量，则可以使用中括号（[]）括起来，在中括号之内声明的变量属于局部变量，而中括号之外声明的变量都是全局变量。中括号的使用不仅使配置文件中声明变量的模块变得清晰，而且可以在模板中选择加载中括号内的变量。

2. 加载配置文件

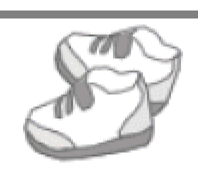
加载配置文件应用 Smarty 的内建函数 config_load，其语法如下：

```
{config_load file="file_name " section="add_attribute" scope="" global=""}
```

参数说明：

❶ file：指定包含的配置文件的名称。







❷ section：附加属性，当配置文件中包含多个部分时应用，指定具体从哪一部分中取得变量。

❸ scope：加载数据的作用域，取值必须为 local、parent 或 global。local 说明该变量的作用域为当前模板；parent 说明该变量的作用域为当前模板和当前模板的父模板（调用当前模板的模板）；global 说明该变量的作用域为所有模板。

❹ global：说明加载的变量是否全局可见，等同于 scope=parent。

提示：

当指定 scope 属性时，可以设置 global 属性，但模板忽略该属性值，而以 scope 属性为准。

3. 引用配置文件中的变量

配置文件加载成功后，就可以在模板中引用配置文件中声明的变量。引用配置文件应用的是“#”或者 Smarty 的保留变量$smarty.config。其应用示例如下：

```
{ config_load file="file_con.conf"}                {* 加载配置文件 *}
{#title#}
    <td height="228" colspan="2" align="left" valign="top" class="{$smarty.config.styles}">
```

4. 应用示例

（1）创建 system 文件夹，定义 Smarty 文件夹，存储编译目录、缓存目录和配置文件目录；创建类文件 system.smarty.inc.php，定义 Smarty 的配置类，在该类中指定配置文件存储的目录，定义类的实例化文件 system.inc.php，完成 Smarty 类的实例化操作，并返回操作对象。

（2）创建 index.php 文件，载入类的实例化文件；通过文件系统函数 file_get_contents()读取文本文件中的数据，并且将数据存储到模板变量中，最终指定模板页。其代码如下：

```
<?php
header ( "Content-type: text/html; charset=UTF-8" );        //设置文件编码格式
include("system/system.inc.php");                           //包含配置文件
$str = file_get_contents('files/content.txt');              //读取文本文件中的数据
$smarty->assign('content',iconv("gb2312","utf-8",$str));    //将文本文件中的数据存储到模板变量中
$smarty->display('index.html');                             //指定模板页
?>
```

（3）创建 index.html 模板页。首先，通过 config_load()函数加载配置文件。然后，应用“#”和$smarty.config 引用配置文件中的变量。最后，通过模板变量输出文本文件中的数据。其关键代码如下：

```
{ config_load file="file_con.conf"}                       {* 加载配置文件 *}
<title>{#title#}</title>
<body  leftmargin="{#leftmargin#}"  topmargin="{#topmargin#}"  marginwidth="{#marginwidth#}"
marginheight="{#marginheight#}">
   <td height="228" colspan="2" align="left" valign="top" class="{$smarty.config.styles}">{$content} </td>
```

问题 266　在 Smarty 模板中，如何截取中文字符串？

问题阐述

在 PHP 中，可以通过自定义函数来完成对中文字符串的截取，那么在 Smarty 模板中，如



何截取中文字符串呢？



Note

专家解答

在 Smarty 模板中提供了 truncate 方法对字符串进行截取，该方法会截取到一个词的末尾，直接避免了在截取中文字符串时出现乱码的问题。truncate 方法的参数说明如表 14.4 所示。

表 14.4　truncate 方法的参数说明

参 数 位 置	说　明
1	第一个参数，指定截取字符的数量，类型为Integer
2	第二个参数，指定截取后追加在截取词后面的字符串，类型为String
3	第三个参数，类型为Boolean，如果该值为false，表示截取到词的边界；如果该值为true，表示截取时精确到指定字符

应用示例

在本示例中，应用 truncate 方法对论坛中的标题和内容进行截取，并且应用省略号替换截取的内容。其具体步骤如下：

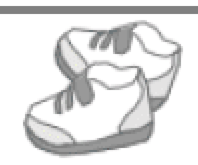
（1）创建 system 文件夹，定义 Smarty 文件夹，存储编译目录、缓存目录；创建类文件 system.smarty.inc.php，定义数据库连接、操作、分页和 Smarty 的配置类，定义类的实例化文件 system.inc.php，完成各个类的实例化操作，并返回操作对象。

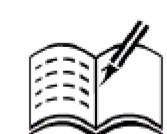
（2）创建 index.php 文件，载入类的实例化文件；分页读取数据库中存储的论坛数据，并且将读取的结果存储到模板变量中，最终指定模板页。其代码如下：

```
<?php
require_once("system/system.inc.php");   //调用指定的文件
$array=$seppage->ShowDate("select*from tb_guestbook order by id desc",$conn,6,$_GET["page"]);
                                         //调用分页类，实现分页功能
if(!$array){
    $smarty->assign("iscommo","F");      //判断如果执行失败，则输出模板变量 iscommo 的值为 F
}else{
    $smarty->assign("iscommo","T");      //判断如果执行成功，则输出模板变量 iscommo 的值为 T
    $smarty->assign("showpage",$seppage->ShowPage("帖子","条","","a1"));
                                         //定义输出分页数据的模板变量 showpage
    $smarty->assign("array",$array);
}
    $smarty->assign('title','truncate 方法截取字符串');            //定义标题
    $smarty->display('index.html');                              //指定模板页
?>
```

（3）创建 index.html 模板页，获取模板变量传递的数据，实现数据的分页输出，并且通过 truncate 方法对输出的标题和内容进行截取操作，标题截取 30 个字节，内容截取 60 个字节，应用省略号补齐截取部分。其关键代码如下：

```
{section name=id loop=$array}
<tr>
<td>
{$array[id].id} {$array[id].title|truncate:30:"…":false};
积分：{$array[id].integral}
时间：{$array[id].createtime|truncate:12:""}<br>
```





```
                {$array[id].content|truncate:60:" …":false}
        </td>
        </tr>
        {/section}
```

运行结果如图 14.14 所示。



图 14.14　应用 truncate 方法截取字符串

专家点评

在 Smarty 模板中应用 truncate 方法截取中文字符串时同样会出现乱码的问题。要想解决这个问题，就要定位到 Smarty 类库 plugins\modifier.truncate.php 文件，重新编辑 smarty_modifier_truncate 方法，将其中截取字符串使用的 substr()函数全部替换为 mb_substr()函数。这样在调用 truncate 方法截取中文字符串时就不会出现乱码的问题了。

问题 267　在 Smarty 中如何进行编码？

问题阐述

在 Smarty 模板中如何进行编码？

专家解答

在 PHP 脚本中应用 urlencode()函数对超链接中传递的参数进行编码，而在 Smarty 模板中提供 escape 函数对字符串进行编码。

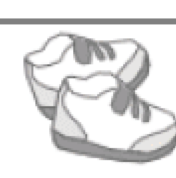
escape 变量用于 HTML 转码、url 转码，在没有转码的变量上转换单引号、十六进制转码或 JavaScript 转码。默认是 HTML 转码。其基本的应用格式如下：

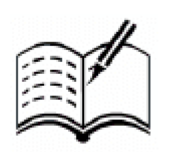
```
{$articleTitle|escape}
{$articleTitle|escape:"html"} {* escapes & " ' < > *}
{$articleTitle|escape:"htmlall"} {* escapes ALL html entities *}
{$articleTitle|escape:"url"}
{$articleTitle|escape:"quotes"}
<a href="mailto:{$EmailAddress|escape:"hex"}">{$EmailAddress|escape:"hexentity"}</a>
```

应用示例

在本示例中应用 escape 函数对 Smarty 模板页中超链接传递的参数进行编码。其关键步骤如下：

（1）在 main.html 模板页中为不同模块之间跳转的超链接传递的参数值进行编码，其关键





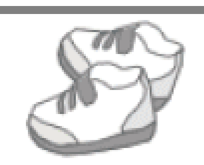
Note

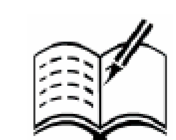
代码如下：

```
<map name="Map" id="Map">
<area shape="rect" coords="29,41,88,62 "href="main.php?caption={"商品添加"|escape:"url"} &amp;
type={"商品管理"}" />
<area shape="rect" coords="30,71,91,90 "href="main.php?caption={"商品修改"|escape:"url"} &amp;
type={"商品管理"}" />
<area shape="rect" coords="31,99,91,118 "href="main.php?caption={"商品删除"|escape:"url"} &amp;
type={"商品管理"}" />
</map>
<map name="Map2" id="Map2">
<area shape="rect" coords="30,45,97,63" href="main.php?caption={"会员删除"|escape:"url"}&amp;
type={"会员管理"}" />
<area shape="rect" coords="34,80,89,98" href="main.php?caption={"会员浏览"|escape:"url"}&amp;
type={"会员管理"}" />
</map>
<map name="Map5" id="Map5">
<area shape="rect" coords="12,11,54,36" href="logout.php" />
</map>
<map name="Map6" id="Map6">
<area shape="rect" coords="14,12,51,36" href="main.php?caption={"商品修改"|escape:"url"}&amp;
type={"商品管理"|escape:"url"}" />
</map>
```

（2）虽然通过 escape 编码后的模板变量输出时是乱码，但是在 switch 语句中仍然可以直接使用$_GET['caption']获取超链接的值，根据这个值作出判断，在主页中输出哪个模块的内容。此时不需要对该参数值进行解码，且需注意的是，执行判断是在 main.php 文件中完成的。其关键代码如下：

```
<?php
session_start();                                          //初始化 SESSION 变量
if($_SESSION['user']!="" and $_SESSION['pass']!=""){      //判断用户名和密码是否为空
require("system/system.inc.php");                         //包含 PHP 配置文件
switch ($_GET['caption']){                                //根据超链接传递的值进行判断
    case "商品添加";
        include "sho_insert.php";                         //包含 PHP 文件
        $smarty->assign('admin_phtml','sho_insert.html');//将 PHP 文件对应的模板文件的名称赋
                                                           给模板变量
    break;                                                //跳出循环
//省略了部分代码
    default:
        include "sho_update.php";
        $smarty->assign('admin_phtml','sho_update.html');
    break;
}
$smarty->assign("title","后台管理系统--".$_GET['caption']);//为模板变量赋值
$smarty->assign("caption",$_GET['caption']);              //为模板变量赋值，输出模块名称
$smarty->assign("type",$_GET['type']);                    //为模板变量赋值，输出模块所属类别
$smarty->assign("user",$_SESSION['user']);                //为模板变量赋值，输出登录用户名称
$smarty->display("main.html");                            //指定模板页
}else{
        echo "<script>alert('您不具备访问权限！'); window.location.href='index.html';</script>";
```





```
    }
    ?>
```

运行结果如图 14.15 所示。



图 14.15　Smarty 模板中对超链接的参数值进行编码

专家点评

经过 escape 变量编码后的字符串在输出时将是一堆乱码，在超链接中通过这种方法可以保证参数值的安全传递。

问题 268　如何在 Smarty 中定义注册模板函数？

问题阐述

如何在 Smarty 中定义注册模板函数？

专家解答

模板函数类似于 PHP 中的自定义函数，特点是其注册在动态页中完成，而实际的应用在静态页中进行，完全体现了 Smarty 的动静分离的开发模式，是在 Smarty 模板中实现一些特殊功能的首选方法。

1. register_function 方法

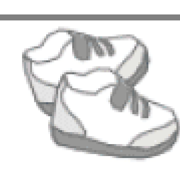
register_function 是 Smarty 模板中的动态注册模板函数插件。其语法如下：

```
    void register_function (string name, mixed impl, bool cacheable, array or null cache_attrs)
```

register_function 方法的参数说明如表 14.5 所示。

表 14.5　register_function 方法的参数说明

参　数	说　明
name	模板函数的名称
impl	执行函数的名称。执行函数的格式可以是一个包含函数名称的字符串；也可以是一个array(&$object, $method)数组形式，其中&$object是一个对象的引用，而$method是它的一个方法；还可以是一个array(&$ class, $method)数组形式，其中$class是一个类的名称，$method是类中的一个方法
cacheable	可选参数，在大多数情况下可以省略
cache_attrs	可选参数，在大多数情况下可以省略







Note

2. Extract()函数

Extract()函数从数组中将变量导入到当前的符号表中。其语法如下：

```
int extract ( array var_array [, int extract_type [, string prefix]] )
```

本函数将变量从数组中导入到当前的符号表中。接受结合数组 var_array 作为参数并将键名当作变量名，值作为变量的值。对每个键/值对都会在当前的符号表中建立变量，并受到 extract_type 和 prefix 参数的影响。其返回值是成功导入到符号表中的变量数目。

说明：

（1）自版本 PHP 4.0.5 起本函数返回被提取的变量数目。（2）EXTR_IF_EXISTS 和 EXTR_PREFIX_IF_EXISTS 是版本 PHP 4.2.0 中引进的。（3）EXTR_REFS 是版本 PHP 4.3.0 中引进的。

Extract()函数检查每个键名，看是否可以作为一个合法的变量名，同时也检查和符号表中已有的变量名的冲突。对待非法数字和冲突的键名的方法将根据 extract_type 参数决定。可以是以下值之一：

☑ EXTR_OVERWRITE：如果有冲突，覆盖已有的变量。
☑ EXTR_SKIP：如果有冲突，不覆盖已有的变量。
☑ EXTR_PREFIX_SAME：如果有冲突，在变量名前加上前缀 prefix。
☑ EXTR_PREFIX_ALL：给所有变量名加上前缀 prefix。自 PHP4.0.5 起这也包括了对数字索引的处理。
☑ EXTR_PREFIX_INVALID：仅在非法数字的变量名前加上前缀 prefix。本标记是 PHP 4.0.5 新加的。
☑ EXTR_IF_EXISTS：仅在当前符号表中已有同名变量时覆盖它们的值，其他的都不处理。可以用在已经定义了一组合法的变量，然后要从一个数组，如$_REQUEST 中提取值覆盖这些变量的场合。本标记是 PHP 4.2.0 新加的。
☑ EXTR_PREFIX_IF_EXISTS：仅在当前符号表中已有同名变量时建立附加了前缀的变量名，其他的都不处理。本标记是 PHP 4.2.0 新加的。
☑ EXTR_REFS：将变量作为引用提取。这有力地表明了导入的变量仍然引用了 var_array 参数的值。可以单独使用这个标志或者在 extract_type 中用 OR 与其他任何标志结合使用。本标记是 PHP 4.3.0 新加的。
☑ 如果没有指定 extract_type，则被假定为 EXTR_OVERWRITE。

注意：

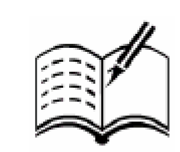
prefix 仅在 extract_type 的值是 EXTR_PREFIX_SAME、EXTR_PREFIX_ALL、EXTR_PREFIX_INVALID 或 EXTR_PREFIX_IF_EXISTS 时需要。如果附加了前缀后的结果不是合法的变量名，将不会导入到符号表中。

3. 应用示例

在本示例中通过注册模板函数对帖子的内容进行截取，如果帖子内容超出指定的长度就对其进行截取，并且使用省略号替代被截取部分。其具体步骤如下：

（1）创建 system 文件夹，定义 Smarty 文件夹，存储编译目录、缓存目录；创建类文件 system.smarty.inc.php，定义数据库连接、操作、分页和 Smarty 的配置类，定义类的实例化文件 system.inc.php，完成各个类的实例化操作，并返回操作对象。





（2）创建 index.php 文件，载入类的实例化文件；编写自定义函数 unHtml，对论坛中的内容进行截取操作；应用 register_function 注册模板函数；分页读取数据库中存储的论坛数据，并且将读取的结

果存储到模板变量中，最终指定模板页。其代码如下：

```
<?php
header ( "Content-type: text/html; charset=UTF-8" );          //设置文件编码格式
require_once("system/system.inc.php");                       //调用指定的文件
function unHtml($params) {                                    //创建模板函数
    extract($params);                                         //读取数据
    if(strlen($text)>40){
        $str=mb_substr($text,0,40,"utf-8")."...";             //截取字符串
    }else{
        $str=$text;
    }
    return $str;                                              //返回截取结果
}
$smarty->register_function("Util", "unHtml");                 //注册模板函数
$array=$seppage->ShowDate("select * from tb_guestbook order by id desc",$conn,6,$_GET["page"]);
                                                              //调用分页类，实现分页功能
if(!$array){
        $smarty->assign("iscommo","F");//判断如果执行失败，则输出模板变量 iscommo 的值为 F
}else{
        $smarty->assign("iscommo","T");//判断如果执行成功，则输出模板变量 iscommo 的值为 T
        $smarty->assign("showpage",$seppage->ShowPage("帖子","条","","a1"));
                                                   //定义输出分页数据的模板变量 showpage
    $smarty->assign("array",$array);
}
    $smarty->assign('title','register_function()方法注册模板函数');
    $smarty->display('index.html');

?>
```

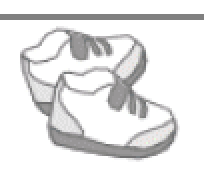
（3）创建 index.html 模板页，获取模板变量传递的数据，实现数据的分页输出，并且调用动态页中注册的模板函数实现对论坛中内容的截取操作。其关键代码如下：

```
{section name=id loop=$array}
<tr>
<td colspan="4" bgcolor="#FFFFFF">{$array[id].id }  {Util text=$array[id].title }积分：
{$array[id].integral}    时间：{$array[id].createtime}<br>{Util text=$array[id].content} </td>
</tr>
{/section}
```

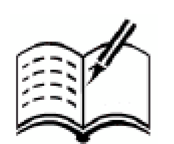
运行结果如图 14.16 所示。



图 14.16　注册模板函数截取字符串



问题 269　如何在 Smarty 中设计页面?



问题阐述

在 Smarty 模板页中，如何对页面进行设计？

专家解答

在 Smarty 模板页中同样可以将网页的头、尾和主文件进行分别存储，从而达到页面重用的效果。

在 Smarty 模板页中应用 include 函数载入模板页。include 函数用于在当前模板中包含其他模板，当前模板中的变量在被包含的模板中可用。include 函数语法如下：

```
{include file="file_name " assign=" " var="    "}
```

参数 file 指定包含模板文件的名称，为必选参数；参数 assign 指定一个变量保存包含模板的输出；参数 var 传递给待包含模板的本地参数，只在待包含模板中有效。

应用示例

本示例中应用 include 函数在模板文件中完成网页结构的设计，实现网页头、尾和主文件的分离。其实现步骤如下：

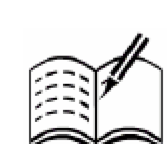
（1）创建 system 文件夹，封装 Smarty 模板的配置方法，这里不再赘述。

（2）创建 index.php 文件，应用 switch 语句根据超链接传递的变量值，包含不同的 PHP 脚本文件，同时将对应的模板页名称赋给指定的模板变量。其代码如下：

```
<?php
require("system/system.inc.php");                          //载入配置文件
switch ($_GET['caption']){                                 //应用 switch 语句，进行判断
    case "reg";
        include "reg.php";                                 //包含 PHP 脚本
        $smarty->assign('admin_phtml','reg.html');         //通过模板变量指定模板页
    break;
    case "log";
        include "log.php";
        $smarty->assign('admin_phtml','log.html');
    break;
    case "from";
        include "for.php";
        $smarty->assign('admin_phtml','for.html');
    break;
    case "mes";
        include "mes.php";
        $smarty->assign('admin_phtml','mes.html');
    break;
    default:
        include "for.php";
        $smarty->assign('admin_phtml','for.html');
    break;
```



```
}
$smarty->assign("title","Smarty 模板中的页面设计--".$_GET[caption]);
$smarty->display("index.html");                    //指定模板页
?>
```



（3）创建 index.html 模板页，应用 include 函数载入页面的头文件（top.html）、尾文件（bottom.html）和主文件。其关键代码如下：

```
{$title}
{include file=top.html}
{include file=$admin_phtml}
{include file=right.html}
{include file=bottom.html}
```

（4）创建 include 函数加载的模板文件和动态 PHP 文件。

运行结果如图 14.17 所示。



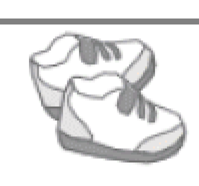
图 14.17　Smarty 模板中的页面设计

专家点评

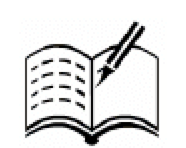
在程序设计中，所有依据条件做出判定的问题都可以用 if 条件语句来解决。不过，在用 if…else 语句处理多个条件的判定问题时，组成条件的表达式在每一个 elseif 中都要计算一次，显得繁琐臃肿。为了避免 if 语句的冗长，提高程序的可读性，可以使用 switch 分支控制语句。switch 语句的语法如下：

```
switch(variable){
    case value1:
        statement1;
        break;
    case value2:
    …
    default:
        default statement n;
}
```

switch 语句根据 variable 的值依次与 case 中的 value 值相比较，如果不相等，继续查找下一个 case；如果相等，就执行对应的语句，直到 switch 语句结束或遇到 break 为止。一般 switch



语句最终都有一个默认值 default，如果在前面的 case 中没有找到相符的条件，则输出默认语句，和 else 语句类似。



问题 270　如何在 Smarty 中制作日历？

问题阐述

如今很多网站都有在线日历功能，如 Google、Yahoo 等。那么在 Smarty 中如何制作日历呢？

专家解答

在 Smarty 模板中制作日历将应用到如下技术：

（1）通过{html_table}函数生成日历表格，{html_table}函数根据指定的列数，生成规定列的表格。其语法如下：

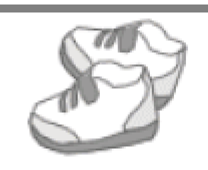
表 14.6　{html_table}函数的语法

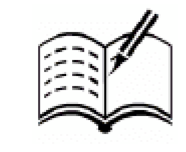
属　性	类　型	是否必须	默认值	描　述
loop	array	Yes	n/a	待遍历的数组
cols	integer	No	3	表格的列数目
table_attr	string	No	border="1"	表格的属性
tr_attr	string	No	empty	行标签属性（或轮转数组）
td_attr	string	No	empty	列标签属性（或轮转数组）
trailpad	string	No		最后一行附加的数据（如果有的话）
hdir	string	No	right	行的对齐方式，可能的值为left或right
vdir	string	No	down	列的对齐方式，可能的值为up或down

自定义函数{html_table}将数组中的数据填充到 HTML 表格中。cols 属性决定表格有多少列；table_attr、tr_attr 和 td_attr 属性决定表格中 tr 和 td 标签的一些附加属性，如果 tr_attr 和 td_attr 属性值为数组，将轮流使用其中的值；如果指定了 trailpad 属性，将在表尾最后一行附加一些数据。

（2）在动态 PHP 文件中通过自定义函数生成日历数据，并且将生成的数据通过模板变量传递到模板页中，在{html_table}函数生成的表格中输出日历数据。自定义函数的语法如下：

```
function get_total_day($date){                          //自定义函数
    $time_stamp=strtotime($date);                       //获取当前时间的时间戳
    $month_ar=explode("/",$date);                       //对当前时间进行拆分
    $month=$month_ar[0];                                //将拆分结果直接的月份值赋给指定变量
    $year=$month_ar[2];                                 //将拆分结果值赋给指定的年份变量
    echo $year ."-". $month;                            //输出当前的年、月值
    $days=date("j",mktime(0,0,0,$month+1,0,$year));     //获取当前时间中本月的天数
    for($i=$days; $i<33; $i++){                         //输出本月的天数
        if(!checkdate($month,$i,$year)){                //判断时间是否合理
            return($i-1);                               //输出时间值
        }
    }
}
```





应用示例

本示例应用自定义函数和 Smarty 模板中的{html_table}函数在 Smarty 模板页中生成一个日历。其具体步骤如下：

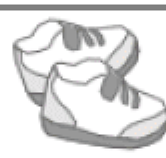
（1）创建 index.php 文件，载入 Smarty 模板的配置文件，封装自定义函数，将生成的日历数据赋给指定的模板变量。其关键代码如下：

```
<?php
require_once("config.php");                          //调用 smarty 文件
$date=date("n/dd/Y");                                //获取当前时间
$week_days=array("Sun"=>0,"Mon"=>1,"Tue"=>2,"Web"=>3,"Thu"=>4,"Fri"=>5,"Sat"=>6,);
                                                     //定义星期数组
$total_day_of_month=get_total_day($date);            //获取当前日期
$starting_day=$week_days[date("D",strtotime($date))]+1; //获取当前日期的周时间
foreach(array_keys($week_days) as $day){             //将周日期对应的字符串存储到数组中
    $days[]=$day;
}
for($i=0;$i<$starting_day;$i++){                     //定义周日期字符串
    $days[]=" ";
}
for($i=1; $i<($total_day_of_month+1);$i++){          //对当前日期描红输出
    if($i==date("d")){
        $days[]="<font color='#FF0000'>".$i."</font>";
    }else{
        $days[]=$i;
    }
}
$smarty->assign("title","在线日历");                 //定义模板变量
$smarty->assign("special_days",$days);               //输出日历数据
$smarty->display("calendar.html");                   //指定模板页
function get_total_day($date){                       //自定义函数
    $time_stamp=strtotime($date);                    //获取当前时间的时间戳
    $month_ar=explode("/",$date);                    //对当前时间进行拆分
    $month=$month_ar[0];                             //将拆分结果直接的月份值赋给指定变量
    $year=$month_ar[2];                              //将拆分结果值赋给指定的年份变量
    echo $year ."-". $month;                         //输出当前的年、月值
    $days=date("j",mktime(0,0,0,$month+1,0,$year));  //获取当前时间中本月的天数
    for($i=$days; $i<33; $i++){                      //输出本月的天数
        if(!checkdate($month,$i,$year)){             //判断时间是否合理
            return($i-1);                            //输出时间值
        }
    }

}
?>
```

（2）在模板文件 calendar.html 中输出模板变量传递的值，通过{html_table}函数生成表格输出当前时间日历。其关键代码如下：

```
<title>{$title}</title>
</head>
<body>
```



```
{html_table loop=$special_days cols=7 }
</body>
</html>
```

运行结果如图 14.18 所示。

2011-8

Sun	Mon	Tue	Web	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

图 14.18　Smarty 模板中的页面设计

专家点评

在应用{html_table}函数生成表格时，不但可以根据指定的列数，生成规定列的表格，而且还可以指定表格颜色及表格附加属性值等。例如：

```
<%{html_table loop=$data}%>
<%{html_table loop=$data cols=4 table_attr='border="0"'}%>
<%{html_table loop=$data cols=4 tr_attr=$tr}%>
```

另外，在本示例中将 Smarty 模板引擎的配置方法封装到 config.php 文件中，在运行示例时，需要配置 SMARTY_PATH 常量的值以及正确地指定项目的存储文件夹，否则将导致程序运行出错。

在本示例中，默认将示例存储于 Apache 服务器的 MR\14 文件夹下，Smarty 模板引擎存储位置是 MR\14\smarty\libs\Smarty.class.php。

问题 271　如何应用 Smarty+PDO 完成数据的分页显示？

问题阐述

Smarty 模板引擎实现网页动静分离，PDO 是数据库抽象层实现对数据库的操作，那么如何应用 Smarty 模板引擎+PDO 数据库抽象层完成数据库中数据的分页输出呢？

专家解答

应用 Smarty+PDO 完成数据的分页显示，在编写分页代码之前需要准备两方面内容：一是 Smarty 类库及所需的各级子目录；二是创建数据库表及插入分页所需要的多条记录。

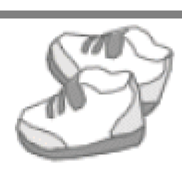
（1）假设程序的主目录为 14，在主目录下创建一个 conn 目录，将 Smarty 类库所在的文件夹 libs 复制到该目录中。在 conn 目录中创建 templates 目录，用来存储程序的模板文件。创建 templates_c 和 cache 两个目录，分别用于存储编译过的模板文件和缓存文件。

（2）假设连接的是 localhost 主机的 MySQL 数据库，用户名和密码分别是“root”和“111”，并且在数据库“db_database”中创建 tb_user 数据表，然后通过 phpMyAdmin 工具向数据表中随意插入多条用户记录，以供分页程序使用。

（3）编译 4 个文件，包括 3 个 PHP 脚本和一个模板文件。每个文件的作用及所存放的目录如表 14.7 所示。

表 14.7　Smarty+PDO 数据分页所用文件解析

文 件 名 称	说　　明
found.database.php	声明数据库操作类、分页操作类所在文件。该文件存放在程序主目录下的conn文件夹中





续表

文 件 名 称	说　　明
connect.database.php	对分页类进行实例化文件。该文件存放在程序主目录下的conn文件夹中
index.php	主索引文件，加载Smarty类并初始化Smarty对象。获取指定页面所需要的数据记录，并发送到模板中。该文件存储在主目录下
index.html	模板文件，以HTML表格形式输出指定页面的记录，调用分页类，实现分页功能。该文件存放在主目录下的子目录templates中

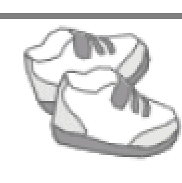
应用 Smarty+PDO 实现数据分页的具体过程如下：

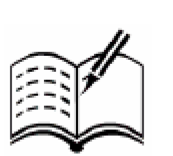
（1）创建 ConnDB 类，定义 ConnDB 方法为成员变量赋值；实现通过 PDO 连接 MySQL 数据库。代码如下：

```
<?php
//数据库连接类
class ConnDB{
	var $dbtype;
	var $host;
	var $user;
	var $pwd;
	var $dbname;
//构造方法
	function ConnDB($dbtype,$host,$user,$pwd,$dbname){
		$this->dbtype=$dbtype;
		$this->host=$host;
		$this->user=$user;
		$this->pwd=$pwd;
		$this->dbname=$dbname;
	}
//实现数据库的连接并返回连接对象
	function GetConnId(){
		if($this->dbtype=="mysql" || $this->dbtype=="mssql"){
			$dsn="$this->dbtype:host=$this->host;dbname=$this->dbname";
		}else{
			$dsn="$this->dbtype:dbname=$this->dbname";
		}
		try {
			$conn = new PDO($dsn, $this->user, $this->pwd);		//初始化一个 PDO 对象，就是
													创建了数据库连接对象$pdo
			$conn->query("set names gb2312");
			return $conn;
		} catch (PDOException $e) {
			die ("Error!: " . $e->getMessage() . "<br/>");
		}
	}
}
```

这就是 ConnDB 类，它存储于示例根目录下 conn 文件夹下的 found.database.php 文件中。

（2）创建 AdminDB 类，定义 ExecSQL 方法，根据传递的 SQL 语句执行不同的操作，例如，如果是 select，则执行查询操作，返回查询结果；如果是 insert 或 update，则直接判断是否





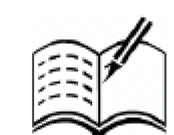
执行成功。代码如下：

```
//数据库管理类
class AdminDB{
    function ExecSQL($sqlstr,$conn){
        $sqltype=strtolower(substr(trim($sqlstr),0,6));
        $rs=$conn->prepare($sqlstr);                    //准备查询语句
        $rs->execute();                                 //执行查询语句并返回结果集
        if($sqltype=="select"){
            $array=$rs->fetchAll(PDO::FETCH_ASSOC);     //获取结果集中的所有数据
            if(count($array)==0 || $rs==false)
                return false;
            else
                return $array;
        }elseif ($sqltype=="update" || $sqltype=="insert" || $sqltype=="delete"){
            if($rs)
                return true;
            else
                return false;
        }
    }
}
```

（3）创建 SepPage 类，定义 ShowData 方法，执行查询语句，将查询结果返回到一个数组中；定义 ShowPage 方法，创建分页超链接，返回值为超链接的字符串。其关键代码如下：

```
//分页类
class SepPage{
    var $rs;
    var $pagesize;
    var $nowpage;
    var $array;
    var $conn;
    var $sqlstr;
    function ShowData($sqlstr,$conn,$pagesize,$nowpage){     //定义方法
        if(!isset($nowpage) || $nowpage=="")                 //判断变量值是否为空
            $this->nowpage=1;                                //定义每页起始页
        else
            $this->nowpage=$nowpage;
        $this->pagesize=$pagesize;                           //定义每页输出的记录数
        $this->conn=$conn;                                   //连接数据库返回的标识
        $this->sqlstr=$sqlstr;                               //执行的查询语句
        $offset=($this->nowpage-1)*$this->pagesize;
        $sql=$this->sqlstr." limit $offset, $this->pagesize";
        $result=$this->conn->prepare($sql);                  //准备查询语句
        $result->execute();                                  //执行查询语句，并返回结果集
        $this->array=$result->fetchAll(PDO::FETCH_ASSOC);//获取结果集中的所有数据
        if(count($this->array)==0 || $this->array==false)
            return false;
        else
            return $this->array;
    }
```





```
        function ShowPage($contentname,$utits,$anothersearchstr,$anothersearchstrs,$class){
            $str="";
            $res=$this->conn->prepare($this->sqlstr);               //准备查询语句
            $res->execute();                                        //执行查询语句，并返回结果集
            $this->array=$res->fetchAll(PDO::FETCH_ASSOC);  //获取结果集中的所有数据
            $record=count($this->array);                            //统计记录总数

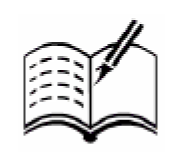
            $pagecount=ceil($record/$this->pagesize);               //计算共有几页
            $str.=$contentname." ".$record." ".$utits." 每页 ".$this->pagesize.
" ".$utits." 第 ".$this->nowpage." 页/共 ".$pagecount." 页";
            $str.="    ";
            if($this->nowpage!=1)
                $str.="<a  href=".$_SERVER['PHP_SELF']."?page=1&page_type=".$anothersearchstr."
&parameter2=".$anothersearchstrs." class=".$class.">首页</a>";
            else
                $str.="<font color='#555555'>首页</font>";
            $str.=" ";
            if($this->nowpage!=1)
                $str.="<a
href=".$_SERVER['PHP_SELF']."?page=".($this->nowpage-1)."&page_type=".$anothersearchstr."&parameter2=".
$anothersearchstrs." class=".$class.">上一页</a>";
            else
                $str.="<font color='#555555'>上一页</font>";
            $str.=" ";
            if($this->nowpage!=$pagecount)
                $str.="<a
href=".$_SERVER['PHP_SELF']."?page=".($this->nowpage+1)."&page_type=".$anothersearchstr."&parameter2="
.$anothersearchstrs." class=".$class.">下一页</a>";
            else
                $str.="<font color='#555555'>下一页</font>";
            $str.=" ";
            if($this->nowpage!=$pagecount)
                $str.="<a href=".$_SERVER['PHP_SELF']."?page=".$pagecount."&page_type=
".$anothersearchstr." &parameter2=".$anothersearchstrs." class=".$class.">尾页</a>";
            else
                $str.="<font color='#555555'>尾页</font>";
            if(count($this->array)==0 || $this->array==false)
                return "无数据！ ";
            else
                return $str;
        }
    }
```

（4）包含 Smarty.class.php 文件，创建 SmartyProject 类，定义 SmartyProject 方法，设置 Smarty 模板文件、编译文件、缓存文件和配置文件的存储路径。

```
    require('conn/libs/Smarty.class.php');                  //包含 Smarty 类文件
    class SmartyProject extends    Smarty{
        function SmartyProject(){
            $this->template_dir = "conn/templates/";          //设置模板文件路径
            $this->compile_dir = "conn/templates_c/";         //设置编译文件路径
```



```
            $this->config_dir = "conn/configs/";             //设置配置文件路径
        }
    }
```



Note

（5）在 conn 文件夹下的 connect.database.php 文件中对分页类进行实例化，返回对象为 $seppage。

（6）在 index.php 文件中调用分页类中的方法，首先判断是否存在缓存文件，如果不存在，则执行数据的分页查询和输出，并将返回结果赋给模板变量；如果存在，则执行缓存文件。其代码如下：

```
<?php
include_once 'conn/connect.database.php';                   //调用类中方法
$arr_page=$seppage->ShowData("select * from tb_user ",$conn,5,$_GET["page"]);  //执行分页查询
if(!$arr_page){
    $smarty->assign("pages","F");
}else{
    $smarty->assign("pages","T");
    $smarty->assign("showpage",$seppage->ShowPage("用户","个","","","a1"));
                                                            //定义分页数据模板变量
    $smarty->assign("arr_pages",$arr_page);                 //将查询的结果赋给模板变量
}
$smarty->display('14.8/index.html');                        //指定模板页
```

（7）在模板文件 index.html 中通过 section 语句循环输出模板变量中传递的数据，通过模板变量输出分页超链接。其代码如下：

```
{section name=id loop=$arr_pages}
  <tr>
    <td height="20" align="center" bgcolor="#FFFFFF">{$arr_pages[id].user}</td>
    <td align="center" bgcolor="#FFFFFF">{$arr_pages[id].pass|truncate:5:"..."}</td>
    <td align="center" bgcolor="#FFFFFF">{$arr_pages[id].dates}</td>
  </tr>
{/section}
  <tr>
    <td height="35" colspan="3" align="center" bgcolor="#FFFFFF">{$showpage}</td>
  </tr>
```

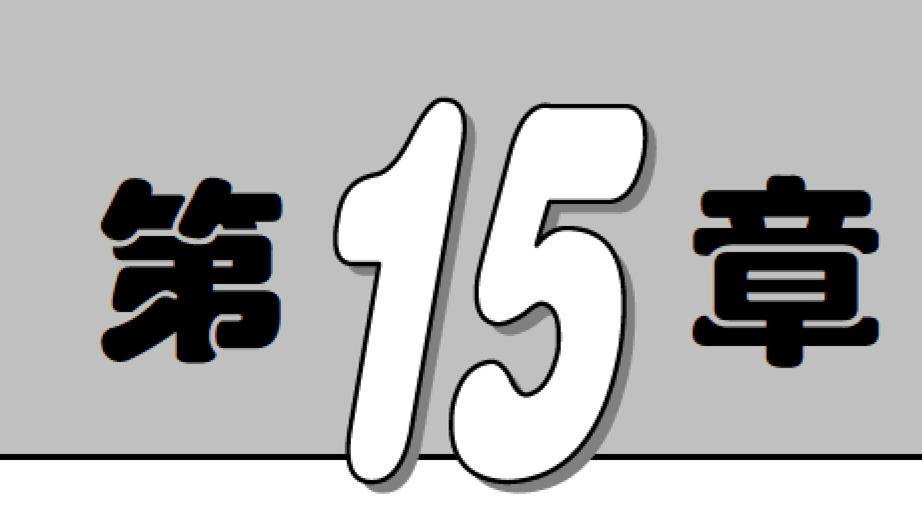
运行结果如图 14.19 所示。

专家点评

在 Smarty 模板页中也可以嵌入 PHP 脚本，其应用的是 Smarty 中的 PHP 标签。PHP 标签中的语句是否被处理取决于$php_handling 的设置。通常情况下是不需要在模板页中嵌入 PHP 脚本的，但是不代表不可以使用。

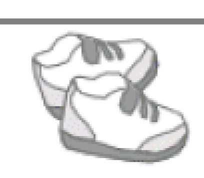


图 14.19　数据的分页输出

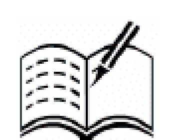


ThinkPHP 框架

- 问题 272　什么是 ThinkPHP？有哪些特点？在哪里下载？
- 问题 273　如何自动生成项目目录？
- 问题 274　ThinkPHP 的命名规范有哪些？
- 问题 275　ThinkPHP 创建项目的基本流程是什么？
- 问题 276　如何应用 ThinkPHP 中的 CURD 操作用户信息？
- 问题 277　如何应用 ThinkPHP 实现用户登录和数据分页输出？
- 问题 278　如何应用验证码类和分页类完成用户登录和分页输出？
- 问题 279　如何通过 ThinkPHP 中扩展类生成中文验证码？
- 问题 280　如何实现传递查询条件的分页？
- 问题 281　如何应用 ThinkPHP 中的扩展类上传文件？
- 问题 282　ThinkPHP 框架中的模板引擎



问题 272　什么是 ThinkPHP？有哪些特点？在哪里下载？



Note

问题阐述

什么是 ThinkPHP？有哪些特点？在哪里下载？

专家解答

1. ThinkPHP 简介

ThinkPHP 可以更方便和快捷地开发和部署应用，其不仅仅是企业级应用，任何 PHP 应用开发都可以从 ThinkPHP 的简单和快速的特性中受益。ThinkPHP 本身具有很多的原创特性，并且倡导大道至简、开发由我的开发理念，用最少的代码完成更多的功能，其宗旨就是让 Web 应用开发更简单、更快速。

ThinkPHP 遵循 Apache2 开源许可协议发布，意味着可以免费使用 ThinkPHP，甚至允许把基于 ThinkPHP 开发的应用开源或商业产品发布/销售。

2. ThinkPHP 框架的特点

ThinkPHP 是一个性能卓越并且功能丰富的轻量级 PHP 开发框架。其宗旨就是让 Web 应用开发更简单、更快速。ThinkPHP 值得推荐的特性包括：

- ☑ 类库导入：ThinkPHP 是首先采用基于类库包和命名空间的方式导入类库，让类库导入看起来更加简单清晰，而且还支持冲突检测和别名导入。为了方便项目的跨平台移植，系统还可以严格检查加载文件的大小写。
- ☑ URL 模式：系统支持普通模式、PATHINFO 模式、REWRITE 模式和兼容模式的 URL 方式，支持不同的服务器和运行模式的部署，配合 URL 路由功能，可以随心所欲地构建需要的 URL 地址和进行 SEO 优化工作。
- ☑ 编译机制：独创的核心编译和项目的动态编译机制，有效减少 OOP 开发中文件加载的性能开销。
- ☑ 查询语言：内建丰富的查询机制，包括组合查询、复合查询、区间查询、统计查询、定位查询、动态查询和原生查询，让查询数据更加简洁高效。
- ☑ 视图模型：轻松动态地创建数据库视图，多表查询不再烦恼。
- ☑ 分组模块：不用担心大项目的分工协调和部署问题，分组模块可以解决跨项目的难题。
- ☑ 模板引擎：系统内建了一款卓越的基于 XML 的编译型模板引擎，支持两种类型的模板标签，融合了 Smarty 和 JSP 标签库的思想，支持标签库扩展。通过驱动还可以支持 Smarty、EaseTemplate、TemplateLite 和 Smart 等第三方模板引擎。
- ☑ Ajax 支持：内置 Ajax 数据返回方法，支持 JSON、XML 和 EVAL 格式返回客户端，并且系统不绑定任何 Ajax 类库，可随意使用自己熟悉的 Ajax 类库进行操作。
- ☑ 缓存机制：系统支持包括文件方式、APC、Db、Memcache、Shmop、Eaccelerator 和 Xcache 在内的多种动态数据缓存类型以及可定制的静态缓存规则，并提供了快捷方法进行存取操作。

3. 环境要求

ThinkPHP 可以支持 Windows/Unix 服务器环境，可运行于包括 Apache、IIS 在内的多种



Web 服务器。需要 PHP 5.0 及以上版本支持。支持 MySQL、MsSQL、PgSQL、Sqlite 和 Oracle 等数据库。



4. 下载 ThinkPHP 框架

ThinkPHP 是一个免费开源、快捷、简单的 OOP 轻量级 PHP 开发框架。它遵循 Apache 2 开源协议发布，是为了敏捷的企业级开发而诞生的。获取 ThinkPHP 的方式有很多。

官方的网站为：http://thinkphp.cn。

SVN 的下载地址为：

完整版本 http://thinkphp.googlecode.com/svn/trunk

核心版本 http://thinkphp.googlecode.com/svn/trunk/ThinkPHP

专家点评

1. 什么是 MVC

MVC 是一种经典的程序设计理念，此模式将应用程序分为 3 个部分：模型层（Model）、视图层（View）、控制层（Controller），MVC 是这 3 个部分英文字母的缩写。

说明：

MVC 设计模式产生的原因是：应用程序中用来完成任务的代码——模型层（也叫"业务逻辑"），通常是程序中相对稳定的部分，重用率高；而与用户交互界面——视图层，却经常改变。如果因需求变动而不得不对业务逻辑代码修改，或者要在不同的模块中应用到相同的功能而重复地编写业务逻辑代码，不仅会降低整体程序开发的进度，而且会使未来的维护变得非常困难。因此，将业务逻辑代码与外观分离，可以更方便地根据需求改进程序，这就是 MVC 设计模式。

在 PHP Web 开发中，MVC 设计模式的各自功能及相互关系如图 15.1 所示。

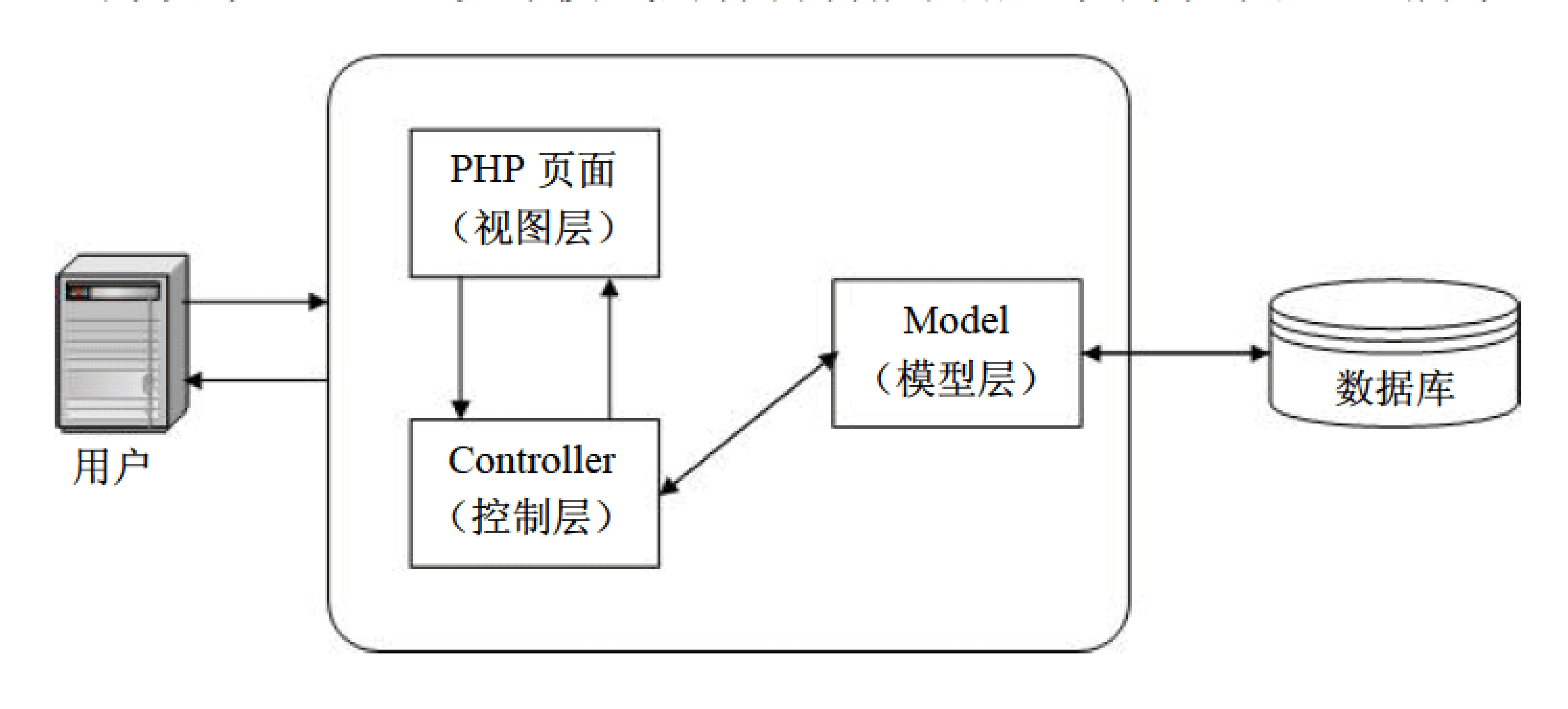


图 15.1 MVC 关系图

☑ 模型层（Model）

模型层是应用程序的核心部分，可以是一个实体对象或一种业务逻辑，它之所以称为模型，是因为它在应用程序中有更好的重用性和扩展性。

☑ 视图层（View）

视图层提供应用程序与用户之间的交互界面，在 MVC 理论之中，这一层并不包含任何的业务逻辑，仅提供一种与用户交互的视图。

☑ 控制层（Controller）

控制层用于对程序中的请求进行控制，其作用就像国家的宏观调控，它可以选择调用哪些

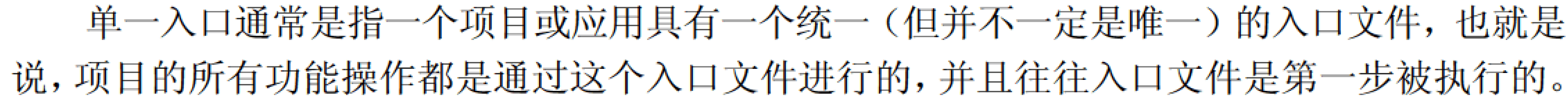


视图或者调用哪些模型。

2. 什么是单一入口



Note



单一入口通常是指一个项目或应用具有一个统一（但并不一定是唯一）的入口文件，也就是说，项目的所有功能操作都是通过这个入口文件进行的，并且往往入口文件是第一步被执行的。

单一入口的好处是项目整体比较规范，因为同一个入口，往往其不同操作之间具有相同的规则。另外一个方面就是单一入口控制较为灵活，因为拦截方便，类似如一些权限控制、用户登录方面的判断和操作可以统一处理。

问题 273　如何自动生成项目目录？

问题阐述

在 ThinkPHP 框架中如何自动生成项目目录？

专家解答

下面通过一个示例讲解在 ThinkPHP 框架中如何自动生成项目目录。项目目录名称为 16.1，其操作步骤如下：

（1）在网站根目录下创建文件夹，并命名为 16.1。

（2）将 ThinkPHP 核心类库存储于 16.1 目录下。

（3）编写入口文件 index.php，将其存储于 16.1 目录下。index.php 文件代码如下：

```
<?php
define('THINK_PATH', 'ThinkPHP');             //定义 ThinkPHP 框架路径（相对于入口文件）
define('APP_NAME', '16.1');                   //定义项目名称
define('APP_PATH', '.');                      //定义项目路径
require(THINK_PATH."/ThinkPHP.php");          //加载框架入口文件
App::run();                                   //实例化一个网站应用实例
?>
```

在运行此文件之前，查看 16.1 项目的文件夹架构，如图 15.2 所示。

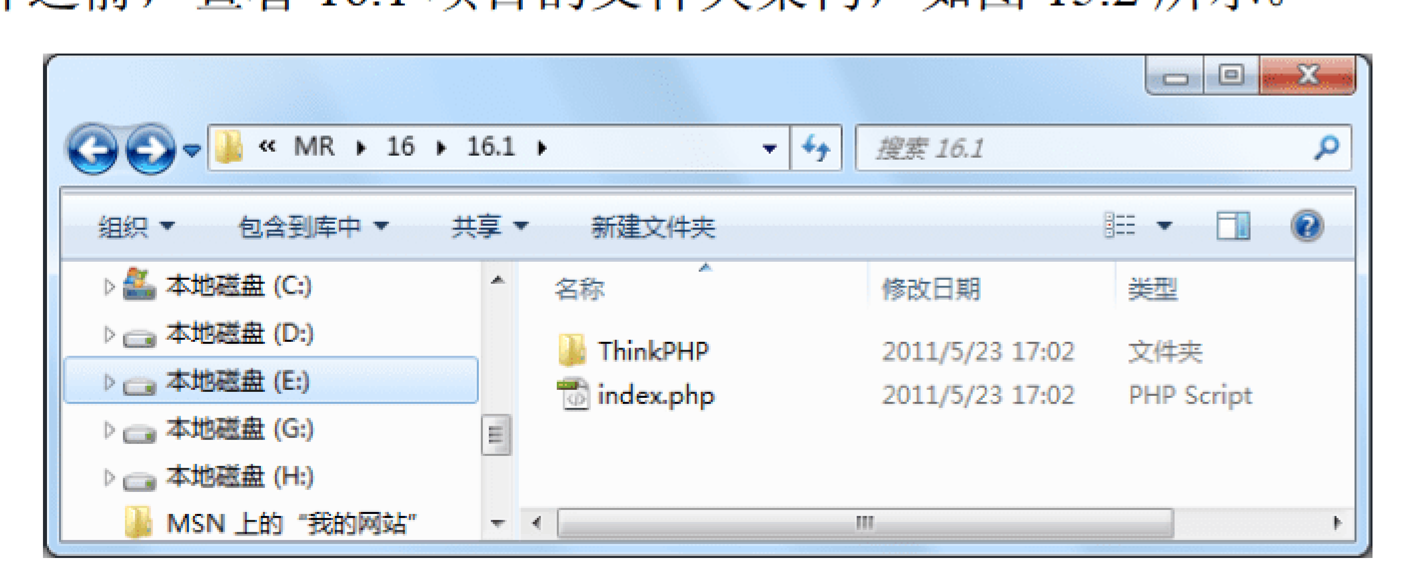


图 15.2　项目文件夹架构

在 IE 浏览器中运行此项目，将输出如图 15.3 所示的运行结果，此为 ThinkPHP 提供的测试内容。再次查看 16.1 项目文件夹，结果显示如图 15.4 所示，在项目根目录下自动生成项目目录。



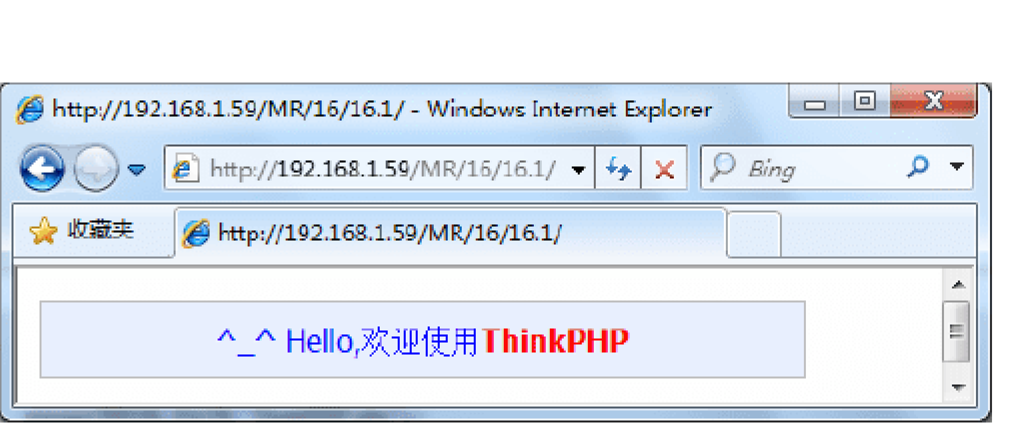


图 15.3　已连接到 ThinkPHP 框架

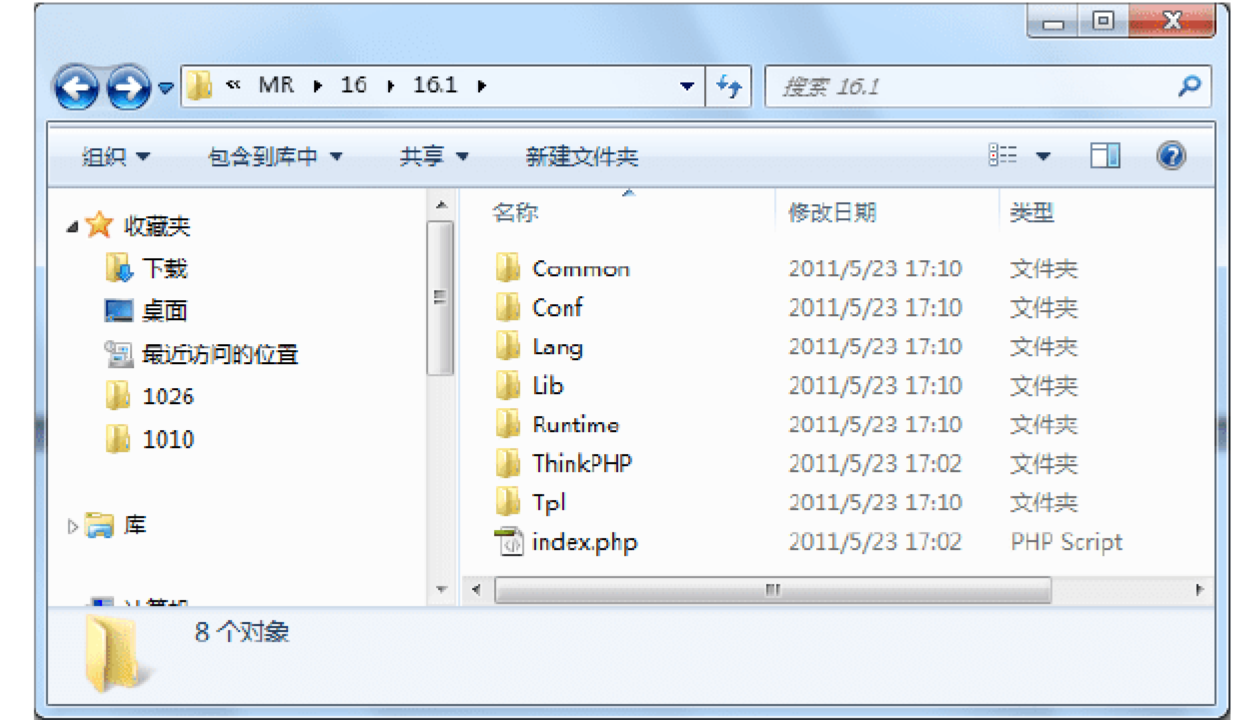


图 15.4　自动生成的项目目录

专家点评

在实际开发过程中，目录结构往往由于项目的复杂而变得复杂。这里向大家推荐两套标准的目录部署方案：方案一如图 15.5 所示。

方案二采用分组模块，如图 15.6 所示。



图 15.5　项目部署方案（一）



图 15.6　项目部署方案（二）

这样部署的好处是系统目录和项目目录可以存储于非 Web 访问目录下，而网站目录下只需放置 Public 公共目录和 index.php 入口文件（如果是多个项目，则每个项目的入口文件都需要放到 Web 目录下），这样就可以大大提高网站的安全性。

问题 274　ThinkPHP 的命名规范有哪些？

问题阐述

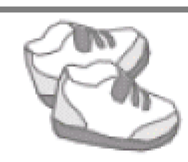
ThinkPHP 框架有其自身的一定规范，要应用 ThinkPHP 框架开发项目，就要尽量遵守它的规范。ThinkPHP 的命名规范有哪些呢？

专家解答

下面就介绍一下 ThinkPHP 的命名规范：

☑ 类文件都是以.class.php 为后缀（这里指的是 ThinkPHP 内部使用的类库文件，不代表外部加载的类库文件），使用驼峰法命名，并且首字母大写，如 DbMysql.class.php。

☑ 函数、配置文件等其他类库文件之外的一般是以.php 为后缀（第三方引入的不做要求）。




- ☑ 确保文件的命名和调用大小写一致，是由于在类 UNIX 系统上，对大小写是敏感的（而 ThinkPHP 在调试模式下，即使在 Windows 平台也会严格检查大小写）。
- ☑ 类名和文件名一致（包括上面说的大小写一致），如 UserAction 类的文件命名是 UserAction.class.php， InfoModel 类的文件名是 InfoModel.class.php。
- ☑ 函数的命名使用小写字母和下划线的方式，如 get_client_ip。
- ☑ Action 控制器类以 Action 为后缀，如 UserAction、InfoAction。
- ☑ 模型类以 Model 为后缀，如 UserModel、InfoModel。
- ☑ 方法的命名使用驼峰法，并且首字母小写，如 getUserName。
- ☑ 属性的命名使用驼峰法，并且首字母小写，如 tableName。
- ☑ 以双下划线“__”打头的函数或方法作为魔法方法，如__call 和__autoload。
- ☑ 常量以大写字母和下划线命名，如 HAS_ONE 和 MANY_TO_MANY。
- ☑ 配置参数以大写字母和下划线命名，如 HTML_CACHE_ON。
- ☑ 语言变量以大写字母和下划线命名，如 MY_LANG；以下划线开头的语言变量通常用于系统语言变量，如_CLASS_NOT_EXIST_。
- ☑ 数据表和字段采用小写加下划线方式命名，如 think_user 和 user_name。

专家点评

在 ThinkPHP 中有一个函数命名的特例，就是单字母大写函数，这类函数通常是某些操作的快捷定义或者有特殊的作用，如 ADSL 方法等，它们有着特殊的含义。另外一点，ThinkPHP 默认使用 UTF-8 编码，所以应确保程序文件采用 UTF-8 编码格式保存，并且去掉 BOM 信息头（去掉 BOM 头信息有很多方式，不同的编辑器都有不同的设置方法，也可以用工具进行统一检测和处理）。

问题 275　ThinkPHP 创建项目的基本流程是什么？

问题阐述

ThinkPHP 具有项目目录自动创建功能，因此构建项目应用程序非常简单，用户只需定义好项目的入口文件，在第一次访问入口文件时，系统就会自动根据在入口文件中所定义的目录路径，迅速创建好项目的相关目录结构。在完成项目目录结构的创建后，接下来需要进行哪些工作才能成功创建项目呢？

专家解答

ThinkPHP 创建项目的基本流程如图 15.7 所示。

应用示例

根据上述讲解的流程创建一个名称为 16.2 的项目，并读取 db_database 数据库中的数据。其操作步骤如下：

（1）创建 db_database 数据库和 think_user 数据表。数据表结构如图 15.8 所示。







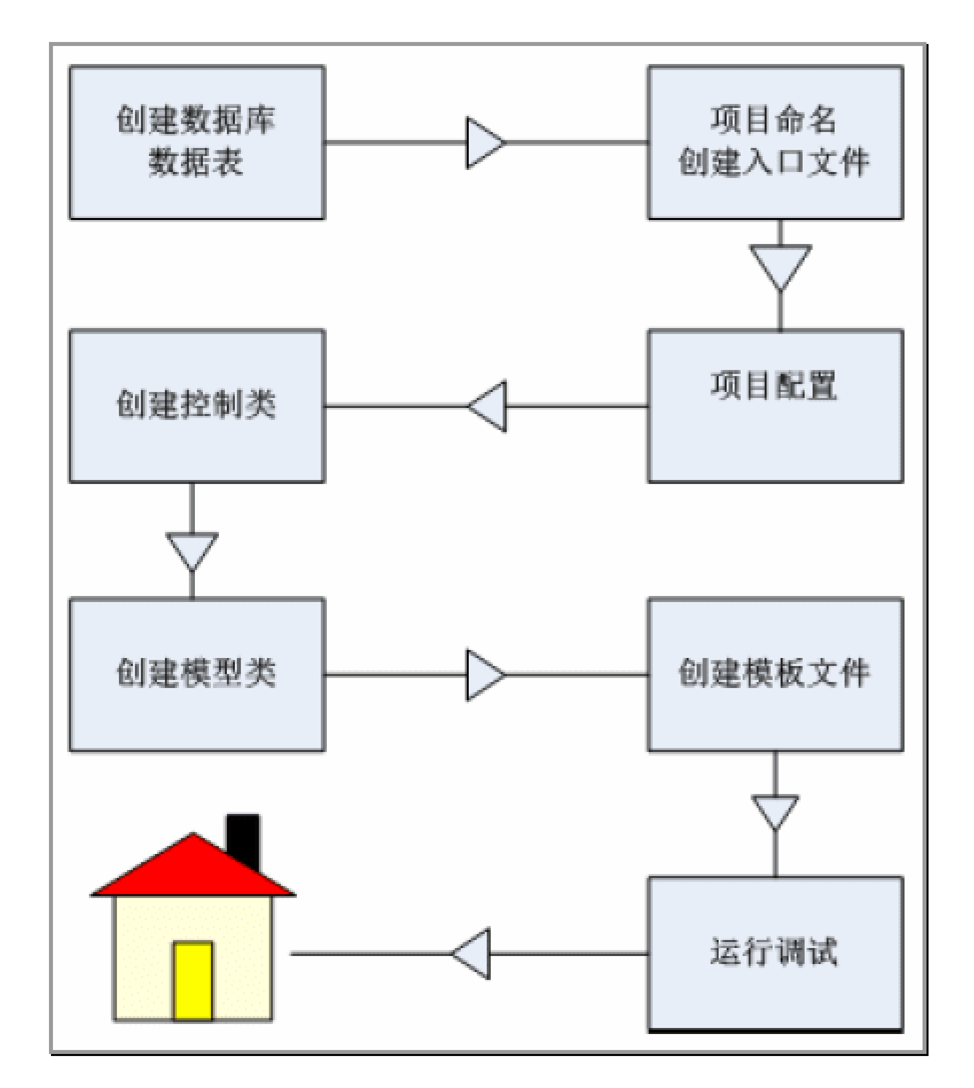
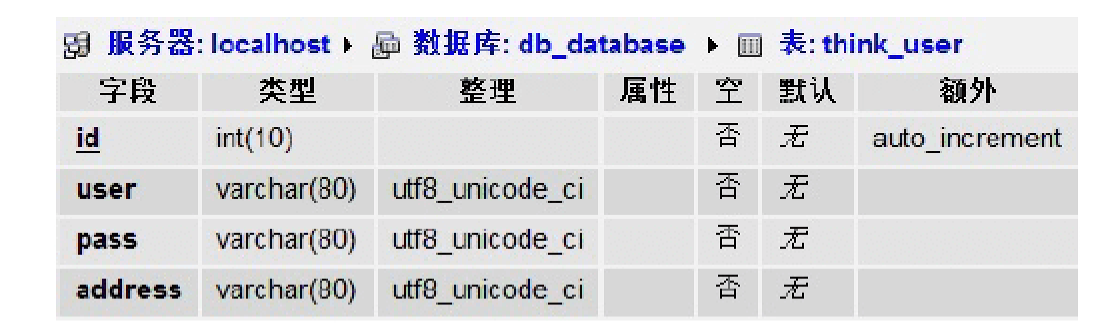


图 15.7　ThinkPHP 创建项目的基本流程



服务器: localhost ▸ 数据库: db_database ▸ 表: think_user

字段	类型	整理	属性	空	默认	额外
id	int(10)			否	无	auto_increment
user	varchar(80)	utf8_unicode_ci		否	无	
pass	varchar(80)	utf8_unicode_ci		否	无	
address	varchar(80)	utf8_unicode_ci		否	无	

图 15.8　数据表结构

（2）载入 ThinkPHP 系统文件，编辑入口文件 index.php，创建名称为 16.2 的项目。index.php 的代码如下：

```
<?php
define('THINK_PATH', '../ThinkPHP/');                //定义 ThinkPHP 框架路径
define('APP_NAME', '16.2');                          //定义项目名称和路径
define('APP_PATH', '.');                             //定义项目名称和路径
require(THINK_PATH."/ThinkPHP.php");                 //加载框架入口文件
App::run();                                          //实例化一个网站应用实例
?>
```

（3）自动生成的项目目录中已经创建了一个空的项目配置文件，位于项目的 Conf 目录下，名称是 config.php。重新编辑此文件，完成数据库的配置。config.php 文件的代码如下：

```
<?php
return array(
    'APP_DEBUG' => true,                             //开启调试模式
    'DB_TYPE'=> 'mysql',                             //数据库类型
    'DB_HOST'=> 'localhost',                         //数据库服务器地址
    'DB_NAME'=>'db_database',                        //数据库名称
    'DB_USER'=>'root',                               //数据库用户名
    'DB_PWD'=>'111',                                 //数据库密码
    'DB_PORT'=>'3306',                               //数据库端口
    'DB_PREFIX'=>'think_',                           //数据表前缀
);
?>
```

（4）在项目的 Lib\Action 目录下定位到自动生成的 IndexAction.class.php 文件，这是 ThinkPHP 的控制器，即 Index 模块。重新编辑控制器的 index 方法，查询指定数据表中的数据，并且完成数据的循环输出。其代码如下：

```
<?php
class IndexAction extends Action{
    public function index() {
```



```
                $db = new Model('user');            //实例化模型类,参数数据表名称，不包含前缀
                $select = $db->select();            //查询数据
                $this->assign('select',$select);    //模板变量赋值
                $this->display();                   //输出模板
        }
}
?>
```

Note

（5）在项目的 Tpl\default 目录下创建 Index 目录，存储 Index 模块的模板文件 index.html，完成数据库中数据的循环输出。其代码如下：

```
<!--循环输出查询结果数据集-->
<volist name='select' id='user' >
ID:{$user.id}<br/>
用户名：  {$user.user}<br/>
地址：  {$user.address}<hr>
</volist>
```

（6）在 IE 浏览器中输入 http://127.0.0.1/ mr/16/example/16.2/，其运行结果如图 15.9 所示。

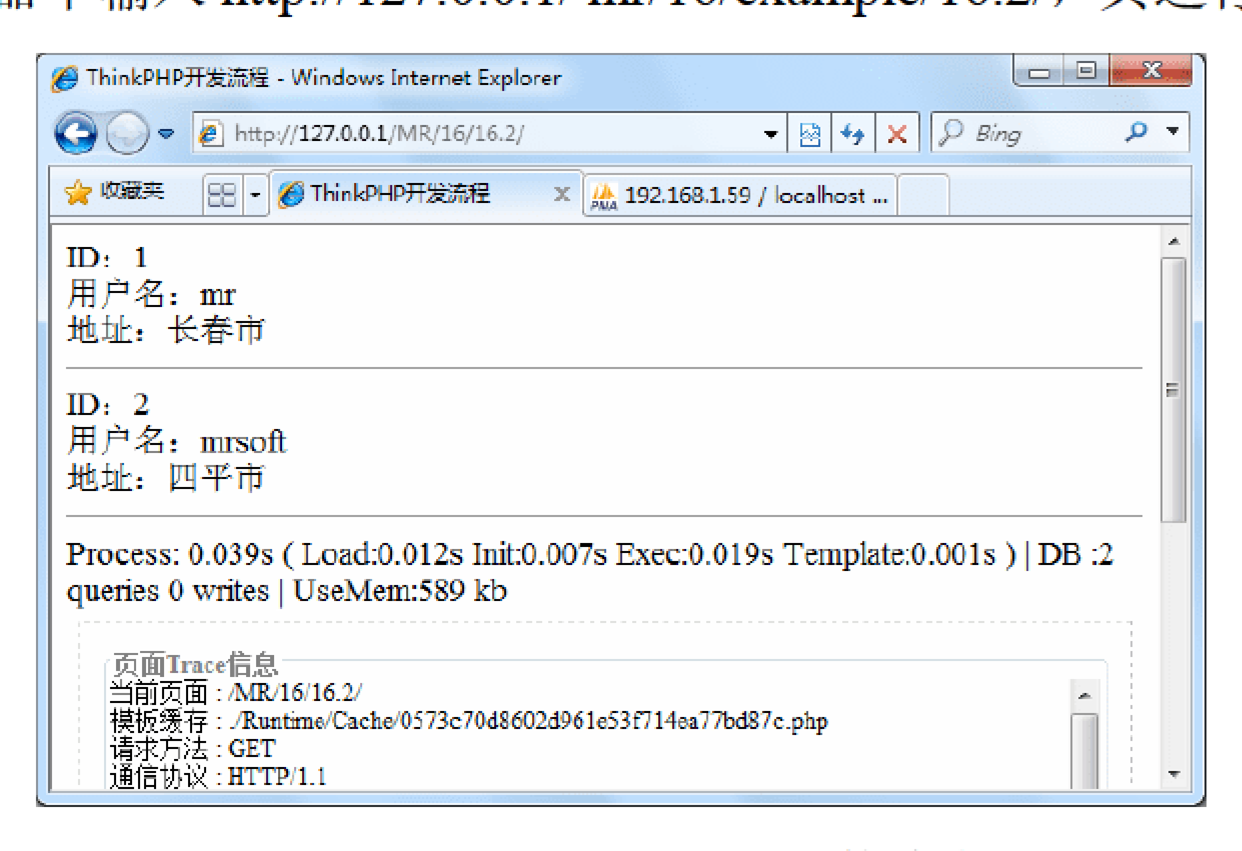


图 15.9　了解 ThinkPHP 项目构建流程

问题 276　如何应用 ThinkPHP 中的 CURD 操作用户信息?

问题阐述

如何应用 ThinkPHP 中的 CURD 操作，实现对用户信息的查询、更新和删除操作？

专家解答

CURD 是数据库操作的缩写词，也是几种数据库操作技术的缩写，“C”代表创建(Create)，“U”代表更新(Update)，“R”代表读取(Read)，“D”代表删除(Delete)。CURD 定义了用于处理数据的基本操作。

CURD 在具体的应用中并非一定使用 create、update、read 和 delete 字样的方法，但是它们完成的功能是一致的。例如，ThinkPHP 就是使用 add、save、select 和 delete 方法表示模型的 CURD 操作。

应用 ThinkPHP 中的 CURD 操作，实现对用户信息的查询、更新和删除操作主要应用到的



技术如下：

（1）通过 M 快捷方法实例化基础模型类。在未定义任何模型时，可以使用下面的方法实例化一个模型类来进行操作：

```
$User = new Model('User');
$User->select();                    //进行其他的数据操作
```

或者使用 M 快捷方法进行实例化，其效果是相同的。

```
$User = M('User');
$User->select();                    //进行其他的数据操作
```

这种方法简单高效，因为不需要定义任何的模型类，所以支持跨项目调用。其缺点是因为没有自定义的模型类，因此无法写入相关的业务逻辑，只能完成基本的 CURD 操作。

（2）连贯操作与 CURD 操作配合应用，完成数据的创建、更新、读取和删除。

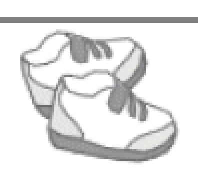
- ☑ Where：用于查询或更新条件的定义。参数支持字符串、数组和对象。
- ☑ Order：对结果进行排序。排序方法支持对多个字段的排序，如 order('status desc,id asc')。order 方法的参数支持字符串和数组，数组的用法如下：order(array('status'=>'desc','id'))。
- ☑ Limit：结果限制。在 ThinkPHP 中，无论操作的是 MySQL、MS SQL Server 还是 Oracle 数据库，其 limit 的方法都是统一的，即 limit('offset,length')。如 limit('1,10')，表示获取从第一条记录开始的 10 条记录。
- ☑ 查询：select()方法的返回值是一个二维数组，如果没有查询到任何结果，则返回一个空的数组。配合上面提到的连贯操作方法可以完成复杂的数据查询。
- ☑ 更新：save()方法在执行更新数据的操作时，如果没有设置任何更新条件，且数据对象本身也不包含主键字段，那么该方法不会更新任何数据库的记录。
- ☑ 删除：delete()方法可以用于删除单个或多个数据，主要取决于删除条件，也就是 where 方法的参数，也可以用 order 和 limit 方法来限制要删除的个数，例如，删除所有状态为 0 的 5 个用户数据按照创建时间排序。代码如下：

```
$User = M("User");                  //实例化 User 对象
$User->where('status=0')->order('create_time')->limit('5')->delete();
```

（3）在控制器中通过 assign()方法为模板变量赋值；通过 display()指定模板页；通过 redirect()方法完成页面的重定向。

- ☑ 通过 assign()方法将控制器中获取的数据赋给模板变量，例如$this->assign('name',$value);。
- ☑ 通过 display()方法指定模板页（display('操作名')）。调用当前模块的其他操作模板。例如，当前是 User 模块下面的 read 操作，而要调用 User 模块的 edit 操作模版，$this->display('edit');。
- ☑ 通过 redirect()方法实现页面的重定向。定义规则分为如下 3 种：第一种基本规则指定项目和模块，第二种数组传参数，第三种不指定项目、模块，即表示当前项目和模块。

```
redirect（'[项目://][路由@][分组名-模块/]操作? 参数 1=值 1[&参数 N=值 N]'）
redirect（'[项目://][路由@][分组名-模块/]操作',array('参数 1'=>'值 1' [,'参数 N'=>'值 N'])）
$this->redirect('Index/index','', 5,'页面跳转中');      //停留 5 秒，跳到 Index 模块 index 操作，显示
                                                        页面跳转字样
```



Note

（4）应用内置模板引擎（ThinkTemplate）中的 foreach 标签在模板页中循环输出模板变量传递的数据。foreach 标签的语法如下：

```
<foreach name="list" item="vo" >
    {$vo.id}
    {$vo.name}
</foreach>
```

foreach 标签用于循环输出，它比 volist 标签简洁，但没有 volist 标签那么多的功能。另外，foreach 标签可以对对象进行遍历输出，而 volist 标签通常是用于输出数组。

应用示例

应用 ThinkPHP 中的 CURD 操作实现对用户信息的查询、更新和删除操作。其具体步骤如下：

（1）创建 253 项目根目录，在根目录下创建项目文件夹 App 和 Public 文件夹存储 CSS、图片和 JS 脚本等文件。

（2）在 253 项目根目录下编辑 index.php 入口文件。其关键代码如下：

```
<?php
define('THINK_PATH', '../ThinkPHP');            //定义 ThinkPHP 框架路径（相对于入口文件）
define('APP_NAME', 'App');                      //定义项目名称
define('APP_PATH', './App');                    //定义项目路径
require(THINK_PATH."/ThinkPHP.php");            //加载框架入口文件
App::run();                                     //实例化一个网站应用实例
?>
```

（3）在 IE 浏览器中运行入口文件，自动生成项目目录。

（4）定位到 App\Conf 目录下，编辑 config.php 文件，并通过 PDO 连接 MySQL 数据库。其代码如下：

```
<?php
return array(
    'DB_TYPE'=> 'pdo',
    //注意 DSN 的配置针对不同的数据库有所区别
    'DB_DSN'=> 'mysql:host=localhost;dbname=db_database',
    'DB_USER'=>'root',
    'DB_PWD'=>'111',
    'DB_PREFIX'=>'think_',
    //其他项目配置参数……
    'APP_DEBUG' => false,                       //关闭调试模式
);
?>
```

（5）定位到 App\Lib\Action\目录下，编写项目控制器。创建 Index 模块，继承系统的 Action 基础类，定义 index()方法，以记录的 ID 值为条件，降幂循环输出 10 条记录。其代码如下：

```
<?php
header("Content-Type:text/html; charset=utf-8");    //设置页面编码格式
class IndexAction extends Action{
    public function index(){
        $db = M('User');                        //实例化模型类，参数数据表名称，不包含前缀
        $select = $db->order('id desc')->limit(10)->select();
```




```
        $this->assign('select',$select);            //模板变量赋值
        $this->display();                           //指定模板页
    }
```

定义 update()方法，首先根据超链接传递 ID 值执行查询，查询出指定的数据，并且将查询结果赋给指定的模板变量。然后，判断表单提交的 ID 值是否存在，如果存在，则以 ID 为条件对指定的数据进行更新操作。其关键代码如下：

```
    public function update(){
        $db = M('User');                            //实例化模型类，参数数据表名称，不包含前缀
        $select = $db->where('id='.$_GET['id'])->select();
        $this->assign('select',$select);            //模板变量赋值
        $this->display(update);                     //指定模板页
        if(isset($_POST['id'])){
            $data['user'] = $_POST['user'];         //要修改的数据对象属性赋值
            $data['pass'] = md5($_POST['pass']);
            $data['address'] = $_POST['address'];
            $result=$db->where('id='.$_POST['id'])->save($data);   //根据条件保存修改的数据
            if($result){
                $this->redirect('Index/index','', 2,'数据更新成功'); //页面重定向
            }
        }

    }
```

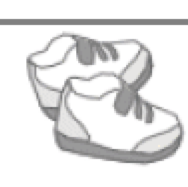
定义 delete()方法，根据超链接传递的 ID 值删除数据库中指定的记录。其关键代码如下：

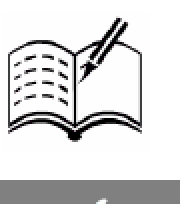
```
    public function delete(){
        $db = M('User');                                         //实例化模型类，参数数据表
                                                                  名称，不包含前缀
        $result=$db->where('id='.$_GET['id'])->delete();         //删除 ID 为 5 的用户数据
        if($result){
            $this->redirect('Index/index','', 2,'数据删除成功');    //页面重定向
        }
    }
}
?>
```

（6）定位到 App\Tpl\default 目录下，创建 Index 模块文件夹。编辑 index 操作的模板文件 index.html，应用 ThinkPHP 内置模板引擎中的 foreach 标签循环输出模板变量传递的数据；创建更新和删除超链接，将指定记录的 ID 作为参数进行传递。其关键代码如下：

```
<foreach name='select' item='user' >
  <tr class="content">
    <td bgcolor="#FFFFFF"> {$user.id}</td>
    <td bgcolor="#FFFFFF"> {$user.user}</td>
    <td bgcolor="#FFFFFF"> {$user.address}</td>
    <td bgcolor="#FFFFFF"><a href="__URL__/update?id={$user.id}">更新</a>/<a href="__URL__/delete?id={$user.id}">删除</a></td>
  </tr>
  </foreach>
```

（7）在 Index 模块文件夹下，编辑 update.html 模板文件，创建表单，将从模板变量中读





取的数据作为表单元素的默认值进行输出，将表单中数据提交到控制器的 update()方法中完成数据的更新操作。其关键代码如下：

```
<form id="form2" name="form2" method="post" action="__URL__/update">
<table width="405" border="1" cellpadding="1" cellspacing="1" bgcolor="#99CC33" bordercolor=
"#FFFFFF">
<foreach name='select' item='user' >
  <tr class="content">
    <td bgcolor="#FFFFFF" class="right" width="103">名称：</td>
    <td bgcolor="#FFFFFF" width="289"> <input type="hidden" name="id" id="hiddenField"
value="{$user.id}" /><input name="user" type="text" id="user" size="20" value="{$user.user}" /></td>
    </tr>
  <tr class="content">
    <td bgcolor="#FFFFFF" class="right">密码：</td>
    <td bgcolor="#FFFFFF"><input name="pass" type="password" id="pass" size="20" value=
"{$user.pass}" />
     </td>
    </tr>
  <tr class="content">
    <td bgcolor="#FFFFFF" class="right"> 地址：</td>
    <td bgcolor="#FFFFFF"> 
      <input name="address" type="text" id="address" size="30" value="{$user.address}" />
    </td>
    </tr>
  <tr class="content">
    <td bgcolor="#FFFFFF"><input type="submit" name="button" id="button" value="更新" /></td>
  </tr>
  </foreach>
</table>
</form>
```

运行结果如图 15.10 所示。

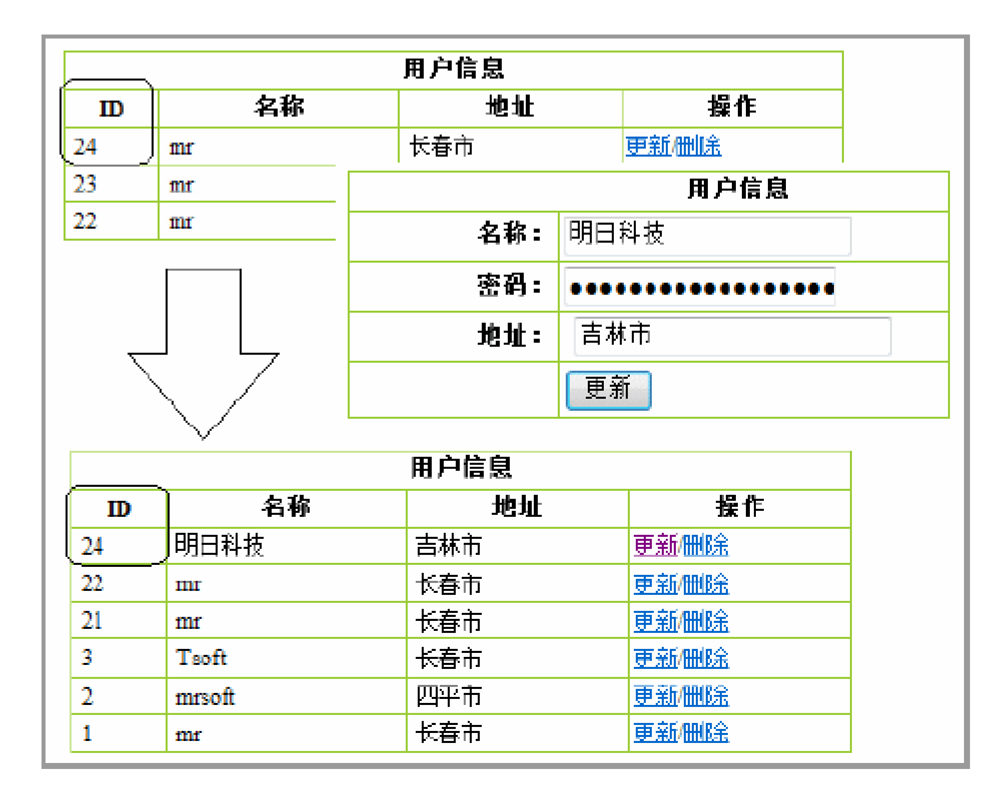
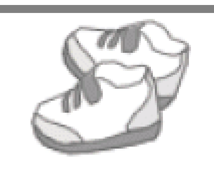


图 15.10　数据更新和删除

专家点评

在 ThinkPHP 的模板文件中有一种特殊的字符串，可以对模板中的目录进行替换，使模板文件的定义更加方便，其默认的替换规则如表 15.1 所示。



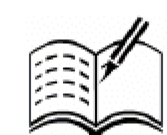


表 15.1 模板中特殊字符串的替换规则

特殊字符串	替 换 描 述
../Public	被替换成当前项目的公共模板目录。通常是：/项目目录/Tpl/default/Public/
__PUBLIC__	被替换成当前网站的公共目录。通常是：/Public/
__TMPL__	替换成项目的模板目录。通常是：/项目目录/Tpl/default/
__ROOT__	会替换成当前网站的地址（不含域名）
__APP__	替换成当前项目的URL地址（不含域名）
__URL__	替换成当前模块的URL地址（不含域名）
__ACTION__	替换成当前操作的URL地址（不含域名）
__SELF__	替换成当前的页面URL

这些特殊的字符串是严格区别大小写的，并且这些特殊字符串的替换规则是可以更改或增加的。只要在项目配置文件中配置 TMPL_PARSE_STRING 即可完成。如果有相同的数组索引，就会更改系统的默认规则。例如：

```
TMPL_PARSE_STRING    => array(
        '__PUBLIC__' => '/Common',                //更改默认的__PUBLIC__替换规则
        '__UPLOAD__' => '/Public/Uploads/',       //增加新的上传路径替换规则
)
```

问题 277　如何应用 ThinkPHP 实现用户登录和数据分页输出？

问题阐述

如何应用 ThinkPHP 实现用户登录和数据分页输出？

专家解答

要实现用户登录和数据分页输出主要应用的技术有两方面：第一个方面是通过 ThinkPHP 框架中的分页类以及连贯操作 Page()方法实现数据的分页输出；第二个方面是应用{$Think.session.session_id|md5}标签输出 SESSION 变量。

（1）通过 import("ORG.Util.Page");载入分页类。

```
import("ORG.Util.Page");                                  //导入分页类
```

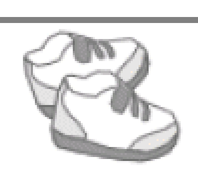
（2）实例化分页类，同时可以传递 3 个参数：第一个是页面总记录数，第二个是每页显示的记录数，第三个为可选参数，通过分页超链接的值。

```
$count = $db->where('address='."'长春市'")->count();       //查询满足要求的总记录数
$Page = new Page($count,1);                  //实例化分页类，传入总记录数和每页显示的记录数
```

（3）调用分页类中的 show()方法，输出分页超链接。

```
$show = $Page->show();                       //分页显示输出
```

（4）Page()：查询分页。属于新增特性，可以更加快速地进行分页查询。Page()方法的用





法和 limit 方法类似，格式为：

```
Page('page[,listRows]')
```

page 表示当前的页数，listRows 表示每页显示的记录数。例如 Page('2,10')，表示每页显示 10 条记录，获取第 2 页的数据。

listRows 如果不写，则会读取 limit('length') 的值，例如 limit(25)->page(3);，表示每页显示 25 条记录，获取第 3 页的数据。如果 limit 也未进行设置，则默认为每页显示 20 条记录。

应用示例

本示例实现用户登录功能，将登录用户的信息存储到 SESSION 变量中，应用 ThinkPHP 中提供的分页扩展类和 Page 方法完成数据的分页输出。其具体步骤如下：

（1）创建 002 项目根目录，在根目录下创建项目文件夹 App 和 Public 文件夹存储 CSS、图片和 JS 脚本等文件。

（2）在 002 项目根目录下编辑 index.php 入口文件。其关键代码如下：

```
<?php
define('THINK_PATH', '../ThinkPHP');                //定义 ThinkPHP 框架路径（相对于入口文件）
define('APP_NAME', 'App');                          //定义项目名称
define('APP_PATH', './App');                        //定义项目路径
require(THINK_PATH."/ThinkPHP.php");                //加载框架入口文件
App::run();                                         //实例化一个网站应用实例
?>
```

（3）在 IE 浏览器中运行入口文件，自动生成项目目录。

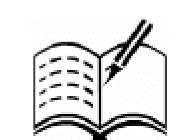
（4）定位到 App\Conf 目录下，编辑 config.php 文件，通过 PDO 连接 MySQL 数据库。其代码如下：

```
<?php
return array(
    'DB_TYPE'=> 'pdo',
    //注意 DSN 的配置针对不同的数据库有所区别
    'DB_DSN'=> 'mysql:host=localhost;dbname=db_database',
    'DB_USER'=>'root',
    'DB_PWD'=>'111',
    'DB_PREFIX'=>'think_',
    //其他项目配置参数………
    'APP_DEBUG' => false,                           //关闭调试模式
);
?>
```

（5）定位到\App\Lib\Action\目录下，编写项目控制器。创建 Index 模块，继承系统的 Action 基础类，定义 index()方法。验证用户提交的用户名和密码是否正确，如果正确，则将登录用户名存储到 SESSION 变量中，并且将网页重定向到 main.html 页面。其代码如下：

```
<?php
session_start();                                    //初始化 SESSION 变量
header("Content-Type:text/html; charset=utf-8");    //设置页面编码格式
class IndexAction extends Action{
    public function index(){
        if(isset($_POST['user'])){
```





```
                if(isset($_POST['user']) && isset($_POST['pass'])){
                    $db = M();                       //实例化模型类，参数数据表名称，不包含前缀
                    $select = $db->query("select * from think_user where user='".$_POST['user']."'
and pass='".$_POST['pass']."'");                      //执行查询语句，验证用户名和密码是否正确
                    if($select){
                        $_SESSION['admin']=$_POST['user'];  //将登录用户名存储到 SESSION 中
                        $this->redirect('Index/main','', 2,'用户 '.$_POST['user'].' 登录成功！');
                                                      //页面重定向
                    }else{
                        $this->redirect('Index/index','', 2,'用户名或者密码不正确！'); //页面重定向
                    }
                }else{
                    $this->redirect('Index/index','', 2,'用户名、密码不能为空！');      //页面重定向
                }
            }
            $this->display();
        }
```

定义 main()方法，载入分页类，完成数据库中数据的分页查询，并且将查询结果赋给模板变量。其代码如下：

```
    public function main(){
        $db = M('User');                   //实例化模型类，参数数据表名称，不包含前缀
        //进行分页数据查询。注意 page 方法的参数的前面部分是当前的页数使用 $_GET[p]获取
        if(isset($_GET['p'])){             //判断分页变量是否存在
            $p=$_GET['p'];
        }else{
            $p=1;
        }
        $list = $db->where('address='.'"长春市"')->order('id desc')->page($p.',1')->select();//查询数据
        $this->assign('select',$list);     //赋值数据集
        import("ORG.Util.Page");           //导入分页类
        $count = $db->where('address='.'"长春市"')->count();       //查询满足要求的总记录数
        $Page = new Page($count,1);        //实例化分页类，传入总记录数和每页显示的记录数
        $show = $Page->show();             //分页显示输出
        $this->assign('page',$show);       //赋值分页输出
        $this->display(main);              //输出模板
    }
```

定义 validatorcode()方法，应用 GD 库中的函数，根据超链接传递的值生成用户登录的验证码。其代码如下：

```
    public function validatorcode(){
        header('content-type:image/png');                                //定义标题 png 格式图像
        $im = imagecreate(65, 25);                                       //定义画布
        imagefill($im, 0, 0, imagecolorallocate($im, 200, 200, 200));    //区域填充
        $validatorCode = $_GET['code'];                                  //获取提交的值
        imagestring($im, rand(3, 5), 10, 3, substr($validatorCode, 0, 1), imagecolorallocate($im, 0,
rand(0, 255), rand(0, 255)));
        imagestring($im, rand(3, 5), 25, 6, substr($validatorCode, 1, 1), imagecolorallocate($im, rand(0,
255), 0, rand(0, 255)));
        imagestring($im, rand(3, 5), 36, 9, substr($validatorCode, 2, 1), imagecolorallocate($im, rand(0,
255), rand(0, 255), 0));
```



```
            imagestring($im, rand(3, 5), 48, 12, substr($validatorCode, 3, 1), imagecolorallocate($im, 0,
rand(0, 255), rand(0, 255)));
            imagepng($im);                                          //生成 PNG 图像
            imagedestroy();                                         //销毁图像
        }
```



Note

（6）定位到 App\Tpl\default 目录下，创建 Index 模块文件夹。编辑 index 操作的模板文件 index.html，载入 CSS 样式文件和 JavaScript 文件，创建表单，完成用户登录信息的提交操作。其关键代码如下：

```
<link href="__ROOT__/Public/Css/style.css" rel="stylesheet" type="text/css" />
<js href="__ROOT__/Public/Js/check.js" />
<form name="form1" method="post"   action="__URL__/index" onSubmit="return chkinput(this)" >
<table width="265" border="0" cellspacing="0" cellpadding="0">
    <tr>
        <td class="title" id="td">用户名：</td>
        <td><input name="user" type="text" size="15" /></td>
    </tr>
    <tr>
        <td class="title" id="td">密码：</td>
        <td><input name="pass" type="password" size="15" /></td>
    </tr>
    <tr>
        <td class="title" id="td">验证码：</td>
        <td>
            <input type="text" name="validatorCode" size="10" />
            <input type="hidden" name="defValidatorCode" value="" />
<script language="javascript">
    var num1=Math.round(Math.random()*10000000);            //生成随机数
    var num=num1.toString().substr(0,4);                    //截取随机数的前 4 个字符
    document.write("<img name=codeimg src='__URL__/validatorcode?code="+num+"'>");
                                                            //将截取值传递到图像处理页中
    form1.defValidatorCode.value=num;                       //将截取值赋给表单中的隐藏域
    function reCode(){                                      //定义方法，重新生成验证码
        var num1=Math.round(Math.random()*10000000);        //生成随机数
        var num=num1.toString().substr(0,4);                //截取随机数
        document.codeimg.src="__URL__/validatorcode?code="+num;
                                                            //将截取值传递到图像处理页中
          form1.defValidatorCode.value=num;                 //将截取值赋给表单中的隐藏域
    }
</script>
            <a href="javascript:reCode()" class="content">看不清</a>
          </td>
    </tr>
</table>
    <input type="image" name="imageField" id="imageField" src="__ROOT__/Public/images/66_05.gif" />
</form>
```

（7）在 Index 模块文件夹下编辑 main.html 文件，通过模板引擎中的 session 标签输出当前登录的用户名，然后通过 foreach 标签循环输出模板变量传递的数据，最后输出模板变量传递的分页超链接。其关键代码如下：



```
        <table width="405" border="1" cellpadding="1" cellspacing="1" bgcolor="#99CC33" bordercolor=
"#FFFFFF">
          <tr>
            <td colspan="3" bgcolor="#FFFFFF" class="title" align="center">当前登录用户：{$Think.session
.admin}</td>
          </tr>
          <foreach name='select' item='user' >
          <tr class="content">
            <td bgcolor="#FFFFFF"> {$user.id}</td>
            <td bgcolor="#FFFFFF"> {$user.user}</td>
            <td bgcolor="#FFFFFF"> {$user.address}</td>
          </tr>
          </foreach>
           <tr class="content">
            <td colspan="3" bgcolor="#FFFFFF"> {$page}</td>
            </tr>
        </table>
```

运行结果如图 15.11 所示。



图 15.11　用户登录和数据的分页输出

专家点评

在利用 Page()方法进行数据的分页查询时，Page()方法的第一个参数是当前的页数，需要使用 $_GET['p']，第二个参数是当前页显示的记录数。

问题 278　如何应用验证码类和分页类完成用户登录和分页输出?

问题阐述

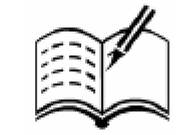
如何应用验证码类和分页类完成用户登录和分页输出？

专家解答

在验证码类中应用 buildImageVerify ()生成验证码。buildImageVerify ()方法的语法如下：

```
buildImageVerify($length,$mode,$type,$width,$height,$verifyName)
```

buildImageVerify()方法的参数说明如表 15.2 所示。



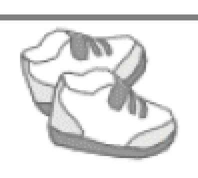




表 15.2　验证码类中 buildImageVerify()的参数说明

参　　数	说　　明
length	验证码的长度，默认为4位数
mode	验证字符串的类型，默认为数字，其他支持类型有0字母；1数字；2大写字母；3小写字母；4中文；5混合（去掉了容易混淆的字符oOLl和数字01）
type	验证码的图片类型，默认为png
width	验证码的宽度，默认会自动根据验证码长度自动计算
height	验证码的高度，默认为22
verifyName	验证码的SESSION记录名称，默认为verify

生成验证码之后，需要在模板页中通过输出生成的验证码图像；在控制器中判断验证码是否正确的代码如下：

```
if($_SESSION['verify'] != md5($_POST['verify'])) {
    $this->error('验证码错误！');
}
```

应用示例

本示例以用户登录和数据的输出为背景，应用 ThinkPHP 中提供的验证码类和分页类生成验证码，完成数据的分页输出。其关键步骤如下：

（1）定位到\App\Lib\Action\目录下，编写项目控制器。创建 Index 模块，继承系统的 Action 基础类，定义 index()方法，验证 SESSION 变量存储的验证码与用户提交的验证码是否相同。验证用户提交的用户名和密码是否正确，如果正确，则将登录用户名存储到 SESSION 变量中，并且将网页重定向到 main.html 页面。其代码如下：

```
<?php
session_start();
header("Content-Type:text/html; charset=utf-8");      //设置页面编码格式
class IndexAction extends Action{
    public function index(){
        if(isset($_POST['user'])){
            if(isset($_POST['user']) && isset($_POST['pass']) && isset($_POST['validatorCode'])){
                if($_SESSION['verify'] == md5($_POST['validatorCode'])) {
                                                      //验证验证码是否正确
                    $db = M();                        //实例化模型类，参数数据表名称，不包含前缀
                    $select = $db->query("select * from think_user where user='".$_POST['user']."'
and pass='".$_POST['pass']."'");                      //执行查询语句，验证用户名和密码是否正确
                    if($select){
                        $_SESSION['admin']=$_POST['user'];
                        $this->redirect('Index/main','', 2,'用户 '.$_POST['user'].' 登录成功！');
                                                                      //页面重定向
                    }else{
                        $this->redirect('Index/index','', 2,'用户名或者密码不正确！');
                                                                      //页面重定向
                    }
                }else{
                    $this->redirect('Index/index','', 2,'验证码不正确！');      //页面重定向
                }
```



```
                }else{
                    $this->redirect('Index/index','', 2,'用户名、密码不能为空！');//页面重定向
                }
            }
            $this->display();
        }
```



定义 main()方法，载入分页类，完成数据库中数据的分页查询，并且将查询结果赋给模板变量。这里应用的是 Page 类和 Limit 方法完成数据的分页输出，其代码如下：

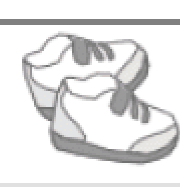
```
    public function main(){
            $db = M('User');                  //实例化模型类，参数数据表名称，不包含前缀
            import("ORG.Util.Page");          //导入分页类
            $count = $db->count();            //统计总记录数
            //$count = $User->where("status=1")->count();          //查询满足要求的总记录数
            $Page = new Page($count,1);       //实例化分页类，传入总记录数和每页显示的记录数
            $show = $Page->show();            //分页显示输出
            //进行分页数据查询 注意 limit 方法的参数要使用 Page 类的属性
            $list = $db->order('id')->limit($Page->firstRow.','.$Page->listRows)->select();
            $this->assign('select',$list);    //赋值数据集
            $this->assign('page',$show);      //赋值分页输出
            $this->display(main);             //输出模板
    }
```

定义 verify()方法，载入 ThinkPHP 中提供的验证码扩展类，调用 buildImageVerify()方法生成验证码。其代码如下：

```
    public function verify(){
        import("ORG.Util.Image");                                    //载入验证码类
        image::buildImageVerify(4,5,'png','65','30','verify');       //生成验证码
    }
```

（2）定位到 App\Tpl\default 目录下，创建 Index 模块文件夹。编辑 index 操作的模板文件 index.html，载入 CSS 样式文件和 JavaScript 文件，创建表单，完成用户登录信息的提交操作，通过 img 标签输出生成的验证码。其关键代码如下：

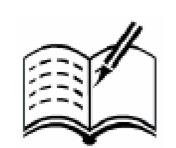
```
    <link href="__ROOT__/Public/Css/style.css" rel="stylesheet" type="text/css" />
    <js href="__ROOT__/Public/Js/check.js" />
    <form name="form1" method="post"    action="__URL__/index" onSubmit="return chkinput(this)" >
    <table width="265" border="0" cellspacing="0" cellpadding="0">
        <tr>
            <td class="title" id="td">用户名：</td>
            <td><input name="user" type="text" size="15" /></td>
        </tr>
        <tr>
            <td class="title" id="td">密码：</td>
            <td><input name="pass" type="password" size="15" /></td>
        </tr>
        <tr>
            <td class="title" id="td">验证码：</td>
            <td><input type="text" name="validatorCode" size="10" /></td>
            <td><img src="__APP__/Index/verify/" /></td>
        </tr>
    </table>
```



```
        </form>
```

注意：

这里应用的验证码是区分字母大小写的。



Note

（3）在 Index 模块文件夹下编辑 main.html 文件，通过模板引擎中的 session 标签输出当前登录的用户名，然后通过 foreach 标签循环输出模板变量传递的数据，最后输出模板变量传递的分页超链接。其关键代码如下：

```
        <table width="405" border="1" cellpadding="1" cellspacing="1" bgcolor="#99CC33" bordercolor=
"#FFFFFF">
            <tr>
                <td colspan="3" bgcolor="#FFFFFF" class="title" align="center">当前登录用户：{$Think.session
.admin}</td>
            </tr>
            <foreach name='select' item='user' >
            <tr class="content">
                <td bgcolor="#FFFFFF"> {$user.id}</td>
                <td bgcolor="#FFFFFF"> {$user.user}</td>
                <td bgcolor="#FFFFFF"> {$user.address}</td>
            </tr>
            </foreach>
             <tr class="content">
                <td colspan="3" bgcolor="#FFFFFF"> {$page}</td>
                </tr>
        </table>
```

运行结果如图 15.12 所示。



图 15.12　验证码类和分页类的应用效果

问题 279　如何通过 ThinkPHP 中的扩展类生成中文验证码？

问题阐述

如何通过 ThinkPHP 中的扩展类生成中文验证码？

专家解答

通过 ThinkPHP 中的扩展类生成中文验证码，其应用的是验证码类中的 GBVerify ()方法。语法如下：

```
        GBVerify ($length,$type,$width,$height,$fontface,$verifyName)
```



GBVerify ()方法的参数说明如表 15.3 所示。

表 15.3　GBVerify ()方法的参数说明

参　　数	说　　明
length	验证码的长度，默认为4位数
type	验证码的图片类型，默认为png
type	验证码的图片类型，默认为png
width	验证码的宽度，默认会自动根据验证码长度自动计算
height	验证码的高度，默认为50
verifyName	验证码的SESSION记录名称，默认为verify
fontface	使用的字体，使用完整文件名或放到图像类所在的目录下，默认使用的字体文件是simhei.tt

注意：

在应用 GBVerify ()方法生成中文验证码时，必须判断以下几个条件：

☑ PHP 是否已经安装 GD 库支持。
☑ 输出之前是否有任何的输出（尤其是 UTF-8 的头信息输出）。
☑ Image 类库是否正确导入。
☑ 如果是中文验证码检查是否有复制字体文件到类库所在目录。

应用示例

下面通过示例讲解如何通过 ThinkPHP 中扩展类生成中文验证码。其关键步骤如下：

在\App\Lib\Action\目录下，在项目控制器的 Index 模块中定义 verify()方法，载入 ThinkPHP 中提供的验证码扩展类，并调用 GBVerify ()方法生成中文验证码。其代码如下：

```
public function verify(){
    import("ORG.Util.Image");                                  //载入验证码类
    image::buildImageVerify(4,5,'png','65','30','verify');     //生成中文验证码
}
```

专家点评

要应用 ThinkPHP 2.1 版本提供的 Image 扩展类完成中文验证码的生成，需要对\Lib\ORG\Util\Image.class.php 文件中 386 行的“msubstr”进行修改，将其修改为“String::msubstr”，否则将无法输出中文验证码。

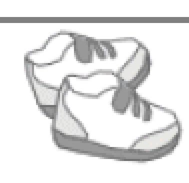
问题 280　如何实现传递查询条件的分页？

问题阐述

如何实现传递查询条件的分页？

专家解答

查询条件在分页超链接中的传递，关键是在实例化 Page 类中第三个参数（$parameter）的



设置，此参数定义的是在分页超链接中传递的参数名称和对应的参数值。其关键代码如下：

```
$Page = new Page($count,1,'key='.$key);
```

第一个参数是查询的总记录数；第二个参数是每页显示的记录数；第三参数是在分页超链接中传递的参数和参数值。



Note

应用示例

本示例通过 TinkPHP 框架开发一个站内搜索的功能，并且对查询结果进行分页输出。其具体步骤如下：

（1）创建 005 项目根目录，在根目录下创建项目文件夹 App 和 Public 文件夹存储 CSS、图片和 JS 脚本等文件。

（2）在 005 项目根目录下编辑 index.php 入口文件。其关键代码如下：

```
<?php
define('THINK_PATH', '../ThinkPHP');              //定义 ThinkPHP 框架路径（相对于入口文件）
define('APP_NAME', 'App');                        //定义项目名称
define('APP_PATH', './App');                      //定义项目路径
require(THINK_PATH."/ThinkPHP.php");              //加载框架入口文件
App::run();                                       //实例化一个网站应用实例
?>
```

（3）在 IE 浏览器中运行入口文件，自动生成项目目录。

（4）定位到 App\Conf 目录下，编辑 config.php 文件，通过 PDO 连接 MySQL 数据库。其代码如下：

```
<?php
return array(
    'DB_TYPE'=> 'pdo',
    //注意 DSN 的配置针对不同的数据库有所区别
    'DB_DSN'=> 'mysql:host=localhost;dbname=db_database',
    'DB_USER'=>'root',
    'DB_PWD'=>'111',
    'DB_PREFIX'=>'think_',
    //其他项目配置参数……
    'APP_DEBUG' => false,                         //关闭调试模式
);
?>
```

（5）定位到\App\Lib\Action\目录下，编写项目控制器。创建 Index 模块，继承系统的 Action 基础类，定义 index()方法。首先，获取查询关键字（通过$_POST 获取表单提交的查询关键字；通过$_GET 获取超链接传递的查询关键字，这是查询结果分页输出的关键），并且将其赋给指定的变量。其代码如下：

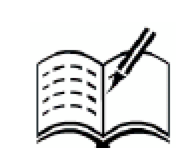
```
<?php
session_start();                                      //初始化 SESSION 变量
header("Content-Type:text/html; charset=utf-8");      //设置页面编码格式
class IndexAction extends Action{
    public function index(){
        if(isset($_POST['key'])){                     //判断查询的关键字是否存在
            $key=$_POST['key'];
```



```
            }else if(isset($_GET['key'])){
                $key=$_GET['key'];
            }
```

然后，实例化基础模型类，载入分页类。其代码如下：

```
    $db = M('User');                                //实例化模型类,参数数据表名称，不包含前缀
    import("ORG.Util.Page");                        //导入分页类
```



接着，判断查询关键字是否存在，如果存在，则根据获取的关键字执行模糊查询，并且对查询结果进行分页输出；如果不存在，则对数据库中的所有数据进行分页输出。其代码如下：

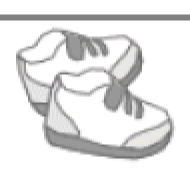
```
    if($key!=''){
        $map="user like('%".$key."%') or address like('%".$key."%') ";
        $list = $db->where($map)->order('id desc')->page($_GET['p'].',1')->select();
        $this->assign('select',$list);              //赋值数据集
        $count = $db->where($map)->count();         //查询满足要求的总记录数
        $Page = new Page($count,1,'key='.$key);     //实例化分页类，传入总记录数、每页显示的
                                                     记录数和查询的关键字
        $show = $Page->show();                      //分页显示输出
        $this->assign('page',$show);                //赋值分页输出
    }else{
        $count = $db->count();              //统计总记录数
        $Page = new Page($count,1);         //实例化分页类，传入总记录数和每页显示的记录数
        $show = $Page->show();              //分页显示输出
        //进行分页数据查询。注意 limit 方法的参数要使用 Page 类的属性
        $list = $db->order('id')->limit($Page->firstRow.','.$Page->listRows)->select();
        $this->assign('select',$list);      //赋值数据集
        $this->assign('page',$show);        //赋值分页输出
    }
```

最后，应用 display()方法指定模板页。其代码如下：

```
            $this->display();               //输出模板
        }
    }
    ?>
```

（6）定位到 App\Tpl\default 目录下，创建 Index 模块文件夹。编辑 index 操作的模板文件 index.html，载入 CSS 样式文；创建表单，提交查询的关键字；应用 foreach 语句完成查询结果的分页输出；应用{$page}输出分页超链接。其关键代码如下：

```
    <link href="__ROOT__/Public/Css/style.css" rel="stylesheet" type="text/css" />
    <form id="form1" name="form1" method="post" action="__URL__/index">
          站内搜索：
          <input type="text" name="key" id="key" />
          <input type="submit" name="button" id="button" value="提交" />
    </form>
    <foreach name='select' item='user' >
      <tr class="content">
        <td bgcolor="#FFFFFF"> {$user.id}</td>
        <td bgcolor="#FFFFFF"> {$user.user}</td>
        <td bgcolor="#FFFFFF"> {$user.address}</td>
      </tr>
```



```
    </foreach>
    <tr class="content">
        <td colspan="3" bgcolor="#FFFFFF"> {$page}</td>
    </tr>
```

运行结果如图 15.13 所示。

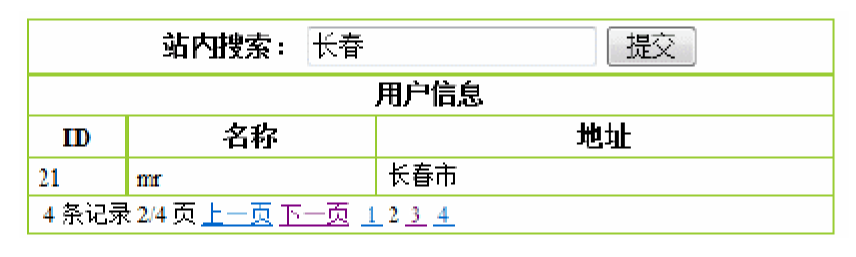


图 15.13　带查询条件的分页

问题 281　如何应用 ThinkPHP 中的扩展类上传文件？

问题阐述

如何应用 ThinkPHP 中的扩展类上传文件？

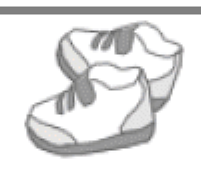
专家解答

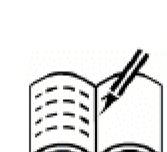
通过 ThinkPHP 中的扩展类上传文件应用的是文件上传扩展类（Lib\ORG\Net\ UploadFile .class.php）中的方法，从而实现文件上传功能。

在 UploadFile 类中可以对上传文件的属性（参数）进行控制，其可控属性如表 15.4 所示。

表 15.4　在 UploadFile 类中上传文件的可控属性

属　　性	描　　述
maxSize	控制上传文件的大小（以字节为单位），默认为-1（不限制大小）
savePath	控制上传文件在服务器中的存储位置，如果设置为空，则使用UPLOAD_PATH常量定义的路径
saveRule	上传文件的保存规则，必须是一个无须任何参数的函数名，如time、uniqid com_create_guid 等，但必须保证生成的文件名是唯一的，默认是uniqid
hashType	传文件的哈希验证方法，默认是md5_file
autoCheck	是否自动检测附件，默认为自动检测
uploadReplace	存在同名文件是否是覆盖
allowExts	允许上传的文件后缀（留空为不限制），使用数组设置，默认为空数组
allowTypes	允许上传的文件类型（留空为不限制），使用数组设置，默认为空数组
thumb	是否需要对图片文件进行缩略图处理，默认为false
thumbMaxWidth	缩略图的最大宽度，多个使用逗号分隔
thumbMaxHeight	缩略图的最大高度，多个使用逗号分隔
thumbPrefix	缩略图的文件前缀，默认为thumb_
thumbSuffix	缩略图的文件后缀，默认为空
thumbPath	缩略图的保存路径，若留空则取文件上传目录本身
thumbFile	指定缩略图的文件名





上传文件的属性设置完成后，应用 upload()方法完成文件上传操作。

文件上传成功后，应用 getUploadFileInfo 方法获取附件信息。其返回值是一个数组，数组中元素的含义如下：

- ☑ key：上传的表单名称。
- ☑ savepath：上传文件的保存路径。
- ☑ name：上传文件的原始名称。
- ☑ savename：上传文件的保存名称。
- ☑ size：上传文件的大小。
- ☑ type：上传文件的 MIME 类型。
- ☑ extension：上传文件的后缀类型。
- ☑ hash：上传文件的哈希验证字符串。

如果上传失败，则应用 getErrorMsg 方法获取错误提示信息。

注意：

在创建文件上传的表单时，在 Form 标签中必须设置 enctype="multipart/form-data"，否则文件不能上传。

应用示例

本示例应用 ThinkPHP 中的文件上传扩展类（Lib\ORG\Net\UploadFile.class.php）实现文件上传的功能，上传 jpg、gif 或 png 格式的图片到服务器指定的文件夹下。其具体步骤如下：

（1）创建 006 项目根目录，在根目录下创建项目文件夹 App 和 Public 文件夹存储 CSS、图片和 JS 脚本等文件。

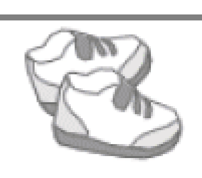
（2）在 006 项目根目录下编辑 index.php 入口文件。其关键代码如下：

```
<?php
define('THINK_PATH', '../ThinkPHP');            //定义 ThinkPHP 框架路径（相对于入口文件）
define('APP_NAME', 'App');                      //定义项目名称
define('APP_PATH', './App');                    //定义项目路径
require(THINK_PATH."/ThinkPHP.php");            //加载框架入口文件
App::run();                                     //实例化一个网站应用实例
?>
```

（3）在 IE 浏览器中运行入口文件，自动生成项目目录。

（4）定位到 App\Conf 目录下，编辑 config.php 文件，并通过 PDO 连接 MySQL 数据库。其代码如下：

```
<?php
return array(
    'DB_TYPE'=> 'pdo',
    //注意 DSN 的配置针对不同的数据库有所区别
    'DB_DSN'=> 'mysql:host=localhost;dbname=db_database',
    'DB_USER'=>'root',
    'DB_PWD'=>'111',
    'DB_PREFIX'=>'think_',
    //其他项目配置参数……
    'APP_DEBUG' => false,                  //关闭调试模式
);
?>
```



（5）定位到\App\Lib\Action\目录下，编写项目控制器。创建 Index 模块，继承系统的 Action 基础类，定义 index()方法，在此方法中没有任何其他操作，直接指定模板页 index.html，在模板页中创建表单，添加文件域，将图片提交到 upload()方法中进行处理。index()方法的代码如下：

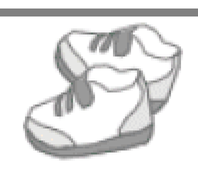
```
public function index(){
    $this->display('index');
}
```

在 Index 控制器的 Index 模块中定义 upload()方法，实现文件上传的功能。首先，载入上传文件类，实例化上传文件类；然后，对上传文件的大小、类型、存储路径进行设置；接着，调用上传文件类中的 upload()方法执行文件的上传操作，如果上传成功，则将上传文件的信息存储到指定的数据表中，并且在 success.html 模板文件中输出上传图片；否则输出上传失败的错误信息，并且跳转到 index.html 页。其完整代码如下：

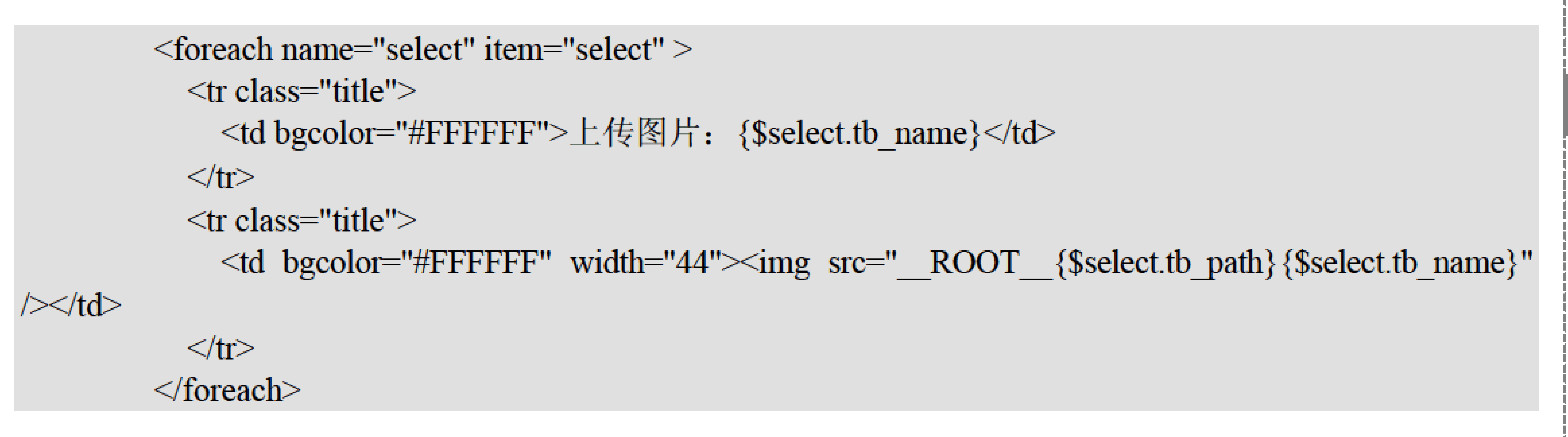
```
public function upload(){
        import("ORG.Net.UploadFile");                          //载入上传文件类
        $upload = new UploadFile();                            //实例化上传类
        $upload->maxSize    = 3145728 ;                        //设置附件上传大小
        $upload->allowExts    = array('jpg', 'gif', 'png', 'jpeg');  //设置附件上传类型
        $upload->savePath =    './Public/Uploads/';           //设置附件上传目录
        if($upload->upload()) {                                //上传成功
            $info =    $upload->getUploadFileInfo();
            $upfile = M("files");                              //实例化 User 对象
            $upfile->create();                                 //创建数据对象
            $upfile->tb_name = $info[0]["savename"];           //保存上传图片名称
            $upfile->tb_path = $info[0]["savepath"];           //保存上传图片路径
            $upfile->tb_date = date("Y-m-d H:i:s");            //保存上传图片时间
            $upfile->add();                                    //写入用户数据到数据库
            $db = M('files');                    //实例化模型类，参数数据表名称，不包含前缀
            $select = $db->order('id desc')->select();         //执行查询语句
            $this->assign('select',$select);                   //模板变量赋值
            $this->success("文件上传成功！");                   //上传成功
        }else{
            $info=$upload->getErrorMsg();                      //上传失败返回错误信息
            $this->redirect('index','', 5,$info);              //页面重定向
        }
    }
```

（6）定位到 App\Tpl\default 目录下，创建 Index 模块文件夹。编辑 index 操作的模板文件 index.html，载入 CSS 样式文；创建表单，定义文件域，将上传图片提交到 upload()方法中进行处理。其关键代码如下：

```
<link href="__ROOT__/Public/Css/style.css" rel="stylesheet" type="text/css" />
<form action="__URL__/upload" method="post" enctype="multipart/form-data" name="form1" id="form1">
  <tr>
    <td colspan="3" bgcolor="#FFFFFF" class="title" align="center">
      选择文件：
        <input name="tb_path" type="file" id="tb_path" size="15" />
      <input type="submit" name="button" id="button" value="提交" />
    </td>
  </tr>
</form>
```



（7）定位到 App\Tpl\default 目录下，创建 Public 模块文件夹。编辑 success 操作的模板文件 success.html。通过 foreach 标签循环输出模板变量传递的数据，包括上传图片的名称以及上传图片在服务器中的存储位置；通过 img 标签根据模板变量传递的名称和路径输出上传的图片。其代码如下：



```
<foreach name="select" item="select" >
  <tr class="title">
    <td bgcolor="#FFFFFF">上传图片：{$select.tb_name}</td>
  </tr>
  <tr class="title">
    <td  bgcolor="#FFFFFF"  width="44"><img  src="__ROOT__{$select.tb_path}{$select.tb_name}"
/></td>
  </tr>
</foreach>
```

运行结果如图 15.14 所示。



图 15.14　文件上传和浏览

问题 282　ThinkPHP 框架中的模板引擎

问题阐述

在 ThinkPHP 框架中实现了动态文件和静态页的分离，动态文件自不必说，那么在静态页模板中是如何完成数据的输出的呢？其应用的又是哪种模板引擎呢？

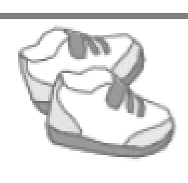
专家解答

ThinkPHP 内置了一个基于 XML 的性能卓越的模板引擎 ThinkTemplate，这是一个专门为 ThinkPHP 服务的内置模板引擎。具体来说，ThinkTemplate 是一个使用 XML 标签库技术编的模板引擎，支持两种类型的模板标签（普通标签和 XML 标签），使用了动态编译和缓存技术，而且支持自定义标签库。

ThinkTemplate 模板引擎生成的编译文件默认存储于 Runtime/Cache 目录下，以模板文件的 md5 编码作为缓存文件名保存。通过 ThinkTemplate 模板引擎中的标签完成对动态文件中数据的各种操作，其常用标签如表 15.5 所示。

表 15.5　ThinkTemplate 模板引擎中的常用标签

标 签 名 称	应 用 描 述
{$name}	输出模板引擎中的变量。注意模板标签的{和$之间不能有任何的空格，否则标签无效





续表

<table>
<tr><th>标 签 名 称</th><th>应 用 描 述</th></tr>
<tr><td>//输出$_SERVER变量
{$Think.server.script_name }
//输出$_SESSION变量
{$Think.session.session_id|md5 }
//输出$_GET变量
{$Think.get.pageNumber }
//输出$_COOKIE变量
{$Think.cookie.name }
{$Think.now } //现在时间
{$Think.template|basename } //模板页面</td><td>系统变量。除了常规变量的输出外，模板引擎还支持系统变量和系统常量以及系统特殊变量的输出。它们的输出不需要事先赋值给某个模板变量。系统变量的输出必须以$Think.打头，并且仍然支持使用函数</td></tr>
<tr><td>{$变量|default="默认值"}</td><td>默认值输出。如果输出的模板变量没有值，但是需要在显示时赋予一个默认值，则可以使用default语法。例如：
{$user.nickname|default="明日科技"}
对系统变量的输出也可以支持默认值，例如：
{$Think.post.name|default="名称为空"}</td></tr>
<tr><td><volist name="list" id="vo" offset="5" length='10'>
{$vo.name}
</volist></td><td>Volist标签主要用于在模板中循环输出数据集或者多维数组。标签参数如下：
Name：表示模板赋值的变量名称，因此不可随意在模板文件中改变。
Id：表示当前的循环变量，可以随意指定，但确保不要和name属性冲突。</td></tr>
<tr><td><volist name="list" id="vo" offset="5" length='10'>
{$vo.name}
</volist></td><td>支持输出部分数据，如输出其中的第5~15条记录。
Offset：表示记录的起始位置。
Length：表示记录的长度。
Mod：控制输出记录的奇偶性，还可以控制在指定的记录换行。例如：
//输出偶数记录
<volist name="list" id="vo" mod="2" >
<eq name="mod" value="1">{$vo.name}</eq>
</volist>
//控制一定记录的换行
<volist name="list" id="vo" mod="5" >
{$vo.name}
<eq name="mod" value="4">
</eq>
</volist></td></tr>
<tr><td><foreach name="list" item="vo" >
{$vo.id}
{$vo.name}
</foreach></td><td>foreach标签用于循环输出，它比volist标签简洁，但没有volist标签那么多的功能。另外，foreach标签可以对对象进行遍历输出，而volist标签通常用于输出数组</td></tr>
</table>

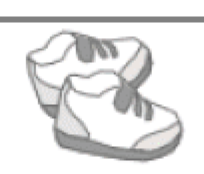
专家点评

上述只是 ThinkTemplate 模板引擎中的一些常用标签，其完整内容读者可以参考 ThinkPHP 手册，由于篇幅所限，这里不再赘述。



PHP 项目开发技巧

- ⏭ 问题 283　如何隐藏 PHP 文件扩展名?
- ⏭ 问题 284　如何在 IIS 服务器中实现 UrlRewrite（伪静态）功能?
- ⏭ 问题 285　如何在 Apache 服务器中实现 UrlRewrite（伪静态）功能?
- ⏭ 问题 286　如何在分页类中实现 UrlRewrite（伪静态）功能?
- ⏭ 问题 287　在 PHP 中如何动态生成静态页面?
- ⏭ 问题 288　如何应用 Zend_mail 发送邮件?
- ⏭ 问题 289　如何解决使用 mail()函数发送邮件时的标题乱码问题?
- ⏭ 问题 290　如何实现带附件的邮件发送?
- ⏭ 问题 291　如何实现邮件群发?
- ⏭ 问题 292　自定义在线编辑器的制作
- ⏭ 问题 293　如何应用 CKEditor 网页编辑器?
- ⏭ 问题 294　如何利用 WebBrowser 打印报表?
- ⏭ 问题 295　如何调用 Word 自动打印指定格式的会议记录?
- ⏭ 问题 296　如何将 Web 页面中的数据导出到 Excel 中并自动打印?
- ⏭ 问题 297　如何将查询结果保存到 Word 中?
- ⏭ 问题 298　如何将查询结果保存到 Excel 中?
- ⏭ 问题 299　如何实现在线支付（工商银行）?
- ⏭ 问题 300　如何实现在线支付（支付宝支付）?





Note

问题 283　如何隐藏 PHP 文件扩展名？

问题阐述

隐藏 PHP 文件扩展名，使其看上去像其他的编程语言，以此提高网站的安全性。下面我们就来探讨一下如何实现。

专家解答

要想隐藏 PHP 文件扩展名，主要是要对以下文件进行修改：

（1）在 php.ini 文件中设置 expose_php=off，可以减少获得的有用信息。

（2）对 Apache 服务器中的配置文件 httpd.conf 进行配置，让类似于 Apache 的 Web 服务端调用 PHP 解释其他扩展名的 PHP 文件，也就是通过设置文件的扩展名来误导攻击者。

☑ 把 PHP 隐藏为另一种语言

```
# 使 PHP 看上去像其他的编程语言
AddType application/x-httpd-php .asp .py .pl
```

☑ 彻底隐藏 PHP

```
# 使 PHP 看上去像不知名的文件类型
AddType application/x-httpd-php .bop 133t
```

专家点评

在完成以上配置之后，要重新启动 Apache 服务器，配置才能够生效，并且要将程序中的 PHP 文件扩展名改为以上配置的扩展名。这种通过隐藏 PHP 来提高安全性的手段虽然防御能力不强，但操作起来较为简单。另外，隐藏 PHP 扩展名还可以使用生成伪静态和真静态页面的方法来实现，不过实现的方法相对要复杂很多。

问题 284　如何在 IIS 服务器中实现 UrlRewrite（伪静态）功能？

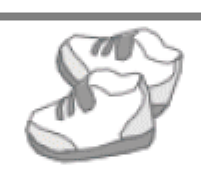
问题阐述

PHP 伪静态技术的应用可以提高 PHP 代码的安全性，便于用户记忆和输入，且易于被搜索引擎收录。例如，http://***/news.php?id=11 可以变为 http://***/news-11.html。这样不但实现开发标记隐藏，避免每个页面都挂上.php 的后缀，而且可以提高网站的安全性。那么在通过 IIS 服务器配置的 PHP 开发环境中如何实现伪静态的功能呢？

专家解答

IIS 默认是不支持伪静态功能的，需要单独安装插件。下面介绍如何让 IIS 支持伪静态的功能，这里以 Windows 2003+IIS6.0 为例进行讲解，具体步骤如下：

第一步：下载 IIS UrlRewrite 插件，到 http://www.helicontech.com/download-isapi_rewrite3.htm，



下载一个 ISAPI_Rewrite 3 Lite installation package（精简版）。

第二步：安装 ISAPI_Rewrite3_0048_Lite.msi。

第三步：配置 IIS 后将其打开，在 Web 属性的“ISAPI 筛选器”中添加位于安装目录的 Helicon\ISAPI_Rewrite3 文件夹中的 ISAPI_Rewrite.dll，名称可以自定义。

至此，IIS 配置完成，下面开发一个 PHP 项目，实现伪静态的功能。

运行位于安装目录 Helicon\ISAPI_Rewrite3 文件夹中的 Helicon Manager.exe，编辑正则表达式。单击“Edit”按钮，输入正则表达式，实现对.php 后缀的替换。其定义的正则表达式如下：

```
RewriteRule ^phpinfo.html$ phpinfo.php
```

运行程序，在浏览器中输入 http://127.0.0.1/phpinfo.html，PHP 程序正常运行，如图 16.1 所示。

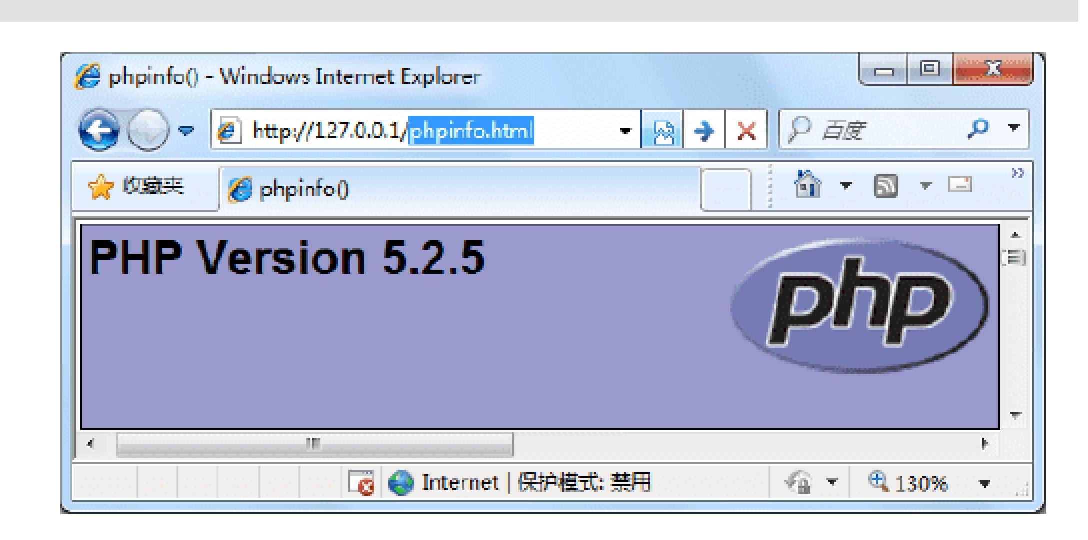


图 16.1　IIS 下的 PHP 伪静态应用

专家点评

在通过 IIS 服务器搭建的 PHP 开发环境中实现伪静态的功能，关键是下载 ISAPI_Rewrite3_0048_Lite.msi 插件和配置 IIS 服务器，还有就是正则表达式的定义，即如何实现后缀及参数的替换。

通过正则表达式对文件的后缀和传递的参数进行匹配，完成对 PHP 文件后缀的隐藏操作。下面介绍了 4 种常用的文件后缀及参数传递的正则表达式编写方法，供读者学习和使用。

```
RewriteRule ^index\.html$ index.php [L]
```

上述直接将文件后缀 php 替换为 html。其替换时的后缀可以任意指定。

```
RewriteRule ^famousword-([0-9]+)-([a-z]+)\.html$ famousword.php?page=$1&flag=$2 [L]
```

上述实现对包含参数的 PHP 文件的后缀进行隐藏，内容分析：

其中 famousword 是文件的名称，“-”是参数直接的分隔符，通过“([0-9]+)”定义第一个参数为数字格式，通过“([a-z]+)”定义第二个参数为字符串格式，定义文件后缀为.html。

“$”符号后面紧跟的是 PHP 文件（famousword.php）以及传递的参数（page=$1&flag=$2）。

“[L]”是结束字符。它实现将 famousword.php 文件隐藏为 famousword.html，并将其参数 page 和 flag 的值分别使用数字和字符串来替换。

```
RewriteRule ^famouswordinfo-([0-9]+)\.html$ famouswordinfo.php?wordid=$1 [L]
```

上述内容实现将 famouswordinfo.php 文件隐藏为 famouswordinfo.html，并将参数 wordid 传递的值使用数字来替换。

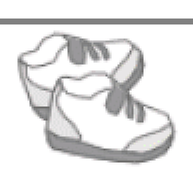
```
RewriteRule ^maindowncenter-(.*)\.html$ maindowncenter.php?a=$1 [L]
```

上述内容中的“(.*)”表示任意字符串，也就是说使用任意字符串来替换参数 a 的值。

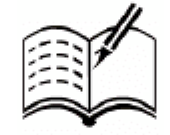
问题 285　如何在 Apache 服务器中实现 UrlRewrite（伪静态）功能？

问题阐述

PHP 伪静态是一个非常不错的技术，通过它的应用可以提高 PHP 代码的安全性，且易于



被搜索引擎收录。那么这项技术是如何实现的呢？确切地说在 Apache 服务器中是如何实现的呢？



专家解答

在 Apache 服务器中实现伪静态功能需要对 Apache 的配置文件进行修改，并且要编译一个.htaccess 文件，在该文件中通过正则表达式实现对文件后缀和传递参数的替换操作。其具体操作步骤如下：

（1）修改 Apache 服务器的配置文件 httpd.conf。该文件存储在 Apache 安装文件夹下的 conf 文件夹中。打开 httpd.conf 文件，定位到如下位置：

```
#LoadModule rewrite_module modules/mod_rewrite.so
```

将该项前面的“#”去掉，启动该项。

（2）查找 httpd.conf 文件，找到其中的“AllowOverride”项，将其值修改为 All。

（3）完成对 httpd.conf 文件的修改后保存该文件，并重新启动 Apache 服务器，使修改生效。

（4）接下来在项目中创建一个.htaccess 文件，实现对 PHP 文件后缀的隐藏操作。.htaccess 文件的格式如下：

```
RewriteEngine on                                        #启动项
```

通过正则表达式对文件的后缀和传递的参数进行匹配，完成对 PHP 文件后缀的隐藏操作。

```
RewriteRule ^index\.html$ index.php [L]
```

应用示例

在站内搜索中应用伪静态技术，对文件的主页和查询结果展示页面使用伪静态技术。在本示例中通过面向对象编程应用 Smarty+PDO 技术完成商品的分页输出，并实现站内查询的功能，最终应用伪静态技术将文件的后缀替换为.html 格式。其编译的.htaccess 文件内容如下：

```
# Helicon ISAPI_Rewrite configuration file
# Version 3.1.0.66
RewriteEngine On
RewriteRule ^index.html$ index.php
RewriteRule ^showcommo.html$ showcommo.php
```

在本示例中，对 Smarty 模板的配置、数据库连接、数据库操作和分页方法进行了封装。然后，实例化类文件，调用类中的方法，完成各种操作。类的实例化文件 system.inc.php 存储于示例的 system 文件夹下，其代码如下：

```
<?php
require("system.smarty.inc.php");                    //载入 Smarty 模板配置类
require("system.class.inc.php");                     //载入数据库连接、操作和分页类文件
$connobj=new ConnDB("mysql","localhost","root","111","db_database",false);   //数据库连接类实例化
$conn=$connobj->GetConnId();
$admindb=new AdminDB();                              //数据库操作类实例化
$seppage=new SepPage();                              //分页实例化
$smarty=new SmartyProject();                         //实例化 Smarty 模板配置类
?>
```

其查询结果如图 16.2 所示。

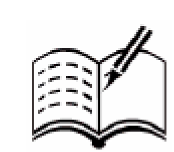




图 16.2　Apache 下的 PHP 伪静态应用

关于示例的完整代码可参考本书配套资源，由于篇幅所限，这里不再赘述。

专家点评

无论是在 IIS 还是 Apache 服务器中实现伪静态功能，最为关键的都是如何定义正则表达式，实现对超链接的替换。即使是在超链接中传递了多个参数，也可以正常地实现转换。例如，对下面的这个有两个参数的超链接实现伪静态功能，其超链接如下：

```
http://127.0.0.1/problem/index.php?page=1&page_type=&parameter2=
```

其定义的正则表达式如下：

```
RewriteRule^index-([0-9]+)-([a-z]+)-([a-z]+)\.html$index.php?page=$1&page_type=$2&parameter2=$3 [L]
```

正则表达式解析：将文件的.php 后缀替换为.html；将参数 page 使用数字进行替换；将参数 page_type 和 parameter2 使用字符串进行替换。

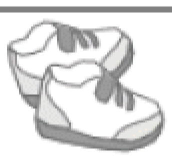
问题 286　如何在分页类中实现 UrlRewrite(伪静态)功能？

问题阐述

前面我们讲解了如何在 IIS 和 Apache 服务器中实现伪静态的功能，那么是否可以在封装的分页类中应用伪静态功能呢？如果可以，那将是一个非常完美的分页类。下面就来讲解这个非常完美的分页类的实现方法。

专家解答

封装一个支持伪静态功能的分页类，首先仍然要创建.htaccess 文件，通过正则表达式实现对超链接的转换操作。定义的.htaccess 文件的内容如下：





```
RewriteEngine On
RewriteRule ^index.phtml$ index.php
RewriteRule ^index-([0-9]+)-([a-z]+)-([A-Z]+)\.phtml$ index.php?page=$1 [L]
RewriteRule ^show_data.phtml$ show_data.php
RewriteRule ^show_data-([0-9]+)\.phtml$ show_data.php?conn_id=$1 [L]
```

然后，封装分页类，其步骤如下：

（1）定义分页类 SepPage，初始化应用的变量。

```
class SepPage{
    var $rs;
    var $pagesize;
    var $nowpage;
    var $array;
    var $conn;
    var $sqlstr;
    var $contentname;
    var $utits;
    var $anothersearchstr;
    var $anothersearchstrs;
    var $class;
```

（2）定义 ShowData()方法，根据参数传递的 SQL 语句（$sqlstr）、连接标识（$conn）、每页显示的记录数（$pagesize）和当前页码值（$nowpage），生成分页查询的 SQL 语句，并且通过 PDO 中的预处理语句执行分页查询，然后通过 PDO 中的 fetchAll 语句获取查询结果集，如果查询结果为空则返回 false，否则返回查询结果集数组。其关键代码如下：

```
    function ShowData($sqlstr,$conn,$pagesize,$nowpage){    //定义方法
        if(!isset($nowpage) || $nowpage=="")                //判断变量值是否为空
            $this->nowpage=1;                               //定义每页起始页
        else
            $this->nowpage=$nowpage;
        $this->pagesize=$pagesize;                          //定义每页输出的记录数
        $this->conn=$conn;                                  //连接数据库返回的标识
        $this->sqlstr=$sqlstr;                              //执行的查询语句
        $offset=($this->nowpage-1)*$this->pagesize;
        $sql=$this->sqlstr." limit $offset, $this->pagesize";
        $result=$this->conn->prepare($sql);                 //准备查询语句
        $result->execute();                                 //执行查询语句，并返回结果集
        $this->array=$result->fetchAll(PDO::FETCH_ASSOC);//获取结果集中的所有数据
        if(count($this->array)==0 || $this->array==false)
            return false;
        else
            return $this->array;
    }
```

（3）定义 ShowPage()方法，根据传递的参数定义分页超链接，实现伪静态链接，最终返回分页超链接字符串。其关键代码如下：

```
    function ShowPage($contentname,$utits,$anothersearchstr,$anothersearchstrs,$class){
        if(!isset($anothersearchstr) || $anothersearchstr=="")  //判断变量值是否为空
            $this->anothersearchstr="a";                        //定义参数值
```



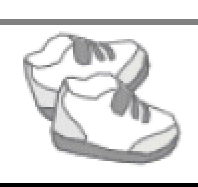


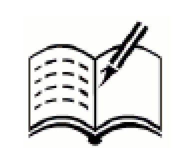
```
                else
                    $this->anothersearchstr=$anothersearchstr;

                if(!isset($anothersearchstrs) || $anothersearchstrs=="")  //判断变量值是否为空
                    $this->anothersearchstrs="A";                         //定义参数值
                else
                    $this->anothersearchstrs=$anothersearchstrs;

                $res=$this->conn->prepare($this->sqlstr);                 //准备查询语句
                $res->execute();                                          //执行查询语句，并返回结果集
                $this->array=$res->fetchAll(PDO::FETCH_ASSOC);            //获取结果集中的所有数据
                $record=count($this->array);                              //统计记录总数
                $pagecount=ceil($record/$this->pagesize);                 //计算共有几页
                $str.=$contentname." ".$record." ".$utits." 每页 ".$this-> pagesize.
" ".$utits." 第 ".$this->nowpage." 页/共 ".$pagecount." 页";
                $str.="    ";
                if($this->nowpage!=1)
                    $str.="<a href=".substr($_SERVER['PHP_SELF'],0,-4)."-"."1"."-".$this->anothersearchstr."-"
.$this->anothersearchstrs.".phtml"." class=".$class.">首页</a>";                 //设置首页超链接
                else
                    $str.="<font color='#555555'>首页</font>";
                $str.=" ";
                if($this->nowpage!=1)
                    $str.="<a href=".substr($_SERVER['PHP_SELF'],0,-4)."-".($this->nowpage-1)."-"
.$this->anothersearchstr."-".$this->anothersearchstrs.".phtml"." class=".$class.">上一页</a>";
                                                                                 //设置上一页的超链接
                else
                    $str.="<font color='#555555'>上一页</font>";
                $str.=" ";
                if($this->nowpage!=$pagecount)
                    $str.="<a href=".substr($_SERVER['PHP_SELF'],0,-4)."-".($this->nowpage+1)
."-".$this->anothersearchstr."-".$this->anothersearchstrs.".phtml"." class=".$class.">下一页</a>";
                                                                                 //设置下一页的超链接
                else
                    $str.="<font color='#555555'>下一页</font>";
                $str.=" ";
                if($this->nowpage!=$pagecount)
                    $str.="<a href=".substr($_SERVER['PHP_SELF'],0,-4)."-".$pagecount."-".$this->
anothersearchstr."-".$this->anothersearchstrs.".phtml"." class=".$class.">尾页</a>";    //设置尾页的超链接
                else
                    $str.="<font color='#555555'>尾页</font>";
                if(count($this->array)==0 || $this->array==false)
                    return "无数据！ ";
                else
                    return $str;
            }
        }
```

至此，分页类创建完成。此时将其存储于 system\system.class.inc.php 文件中。另外，在该文件中还存储了数据库的连接和管理类以及系统常用方法类，由于篇幅所限，这里不再赘述。





Note

说明：

在 ShowPage()方法中定义的超链接就是通过伪静态生成的。通过$_SERVER['PHP_SELF']获取当前脚本文件的名称，然后应用 substr()函数对文件名进行截取，返回没有后缀的文件名称，在配合方法传递的参数，根据伪静态中定义的超链接格式重新定义超链接字符串。

然后在 system 文件夹下创建 system.inc.php 文件，实现类的实例化操作，通过实例化返回的对象调用类中的方法，实现数据库的连接、管理以及数据的分页输出。

运行结果如图 16.3 所示。



图 16.3　在分页类中应用伪静态功能

专家点评

在分页类中实现伪静态是伪静态技术在 IIS 或 Apache 服务器下具体应用的体现，在应用伪静态技术之前必须完成对服务器的配置，然后才可以进行具体的操作。所谓在分页类中应用伪静态技术，其关键就是如何定义通用的分页超链接。如何将这个通用的分页超链接定义成伪静态的格式，其关键就是$_SERVER['PHP_SELF']、substr()函数和字符串连接符的巧妙运用。

问题 287　在 PHP 中如何动态生成静态页面？

问题阐述

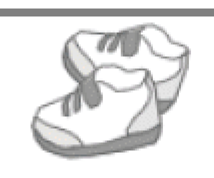
为了有效地提高站点的访问效率、降低服务器负载、提高被搜索引擎搜索到的几率，可以在 PHP 中动态生成静态页面的形式进行输出。那么在 PHP 中如何动态生成静态页面呢？

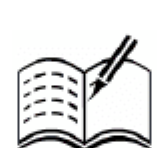
专家解答

PHP 动态生成静态页面主要应用 PHP 中的缓存函数。

（1）ob_start()函数，开启缓存，这时将不会有来自脚本的内容被输出，这些内容都存储在缓存中。语法如下：

```
bool ob_start ( [callback output_callback [, int chunk_size [, bool erase]]] )
```





ob_start()函数的参数说明如表 16.1 所示。

表 16.1　ob_start()函数的参数说明

参　　数	说　　明
output_callback	可选参数。设置回叫函数，当遇到ob_end_flush()函数时或者在脚本末尾缓存自动输出时，回叫函数才被执行。函数会收到以输出的内容作为参数，并且函数需要返回一个包括新内容的变量
chunk_size	可选参数。设置每行输出内容的字节数，达到设置的字节数就会换行
erase	可选参数。如果设置为false，那么在脚本结束后，缓存不会被删除，作用于php 4.3.0之后

（2）ob_end_flush()函数，向浏览器输出缓存中的内容，并关闭缓存，成功返回 true，失败返回 false。

```
bool ob_end_flush ( void )
```

注意：

如果失败将会出现 E_NOTICE 级别的错误信息，返回的布尔值是 php 4.2.0 以后添加的。

（3）ob_get_contents()函数，获取缓存中的内容，失败则返回 false。

```
string ob_get_contents ( void )
```

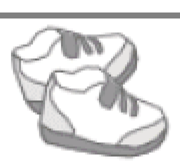
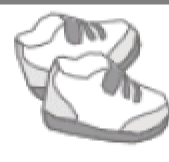
（4）创建 StaticPage 类，应用 ob_start()函数定义 pageBegin()动态页面的开始方法；应用 ob_end_flush()函数定义 pageEnd()动态页面的结束方法；定义 getStaticPageName()方法，根据当前请求地址获取静态页面的名称；定义 makeStaicPage()方法，生成静态页面；定义 reMakeCondition()方法，在指定的时间内重新生成静态页面。

应用示例

应用 PHP 中的缓存函数，创建 StaticPage 类，实现将动态文件生成静态页面。具体实现过程如下：

（1）本示例使用 Adodb 技术管理 MySQL 数据库，为了提高代码重用率、体现 PHP 面向对象编程的特点，创建 ConnDB 数据库连接类实现与 MySQL 数据库的连接，代码如下：

```
<?php
class ConnDB{
        var $dbType;                                    //数据库类型
        var $host;                                      //MySQL 数据库服务器地址
        var $userName;                                  //用户名
        var $password;                                  //密码
        var $dbName;                                    //数据库名
        var $isDebug;                                   //是否调试
        var $connID;                                    //保存数据库连接标识
        function ConnDB ($dbType = 'mysql', $host, $userName, $password, $dbName, $isDebug = false){
            //构造函数，用于对类中的属性进行初始化
            $this->dbType = $dbType;
            $this->host = $host;
            $this->userName = $userName;
            $this->password = $password;
            $this->dbName = $dbName;
```

```
                $this->isDebug = $isDebug;
            }
        function getConnID (){
                require_once 'library/adodb/adodb.inc.php';          //导入 Adodb 类库
                $this->connID = NewADOConnection($this->dbType); //建立连接
                if ($this->dbType == 'mysql' || $this->dbType == 'mssql') { //MySQL 或 SQL Server 数据库
                    $this->connID->Connect($this->host, $this->userName, $this->password, $this->
dbName);
                    if ($this->dbType == 'mysql') {
                        $this->connID->Execute('set names gbk'); //如果是 MySQL 数据库，则设置字符集
                    }
                } elseif ($this->dbType == 'access') {              //Access 数据库
                    $this->connID->Connect('Driver = {Microsoft Access Driver (*.mdb)}; Dbq =' .
realpath($this->dbName) . ';Uid = ' . $this->userName . ';Pwd = ' . $this->password. ';');
                } else {
                    return false;
                }
                $this->connID->debug = $this->isDebug;              //是否调试
                return $this->connID;
        }
        function closeConnID (){                                    //关闭与数据库的连接
                @$this->connID->Disconnect();
        }
    }
```

在 ConnDB 类中不仅可以建立与 MySQL 数据库的连接，而且可以建立与 Access 和 SQL Server 数据库的连接。ConnDB 类中首先使用构造方法对类中的属性进行实例化，然后创建 getConnID()方法用于返回数据库连接的 ID，最后创建 closeConnID()方法释放数据库连接资源。

（2）创建基于 Adodb 类库的分页类 PageDB，用于实现数据的分页显示。PageDB 类的代码如下：

```
    <?php
    class PageDB{
        function pageData ($sql, $connID, $pageSize, $page){
            $arrayPageInfo = array();                       //定义空数组
            if (isset($page) && $page != '') {              //判断页面变量 page 的值是否为空
                $nowPage = $_GET['page'];                   //如果不为空，则获取变量的值
            } else {
                $nowPage = 1;                               //否则为变量赋值为 1
            }
            $rs = $connID->PageExecute($sql, $pageSize, $nowPage);  //执行分页操作
            $arrayData = $rs->GetRows();                    //获取结果
            if (count($arrayData) == 0 || $rs == false) {   //如果结果等于 0 或为 false，则返回 false
                return false;
            } else {
                $arrayPageInfo['data'] = $arrayData;        //将获取的结果定义到数组中
                $rsAll = $connID->Execute($sql);            //执行 SQL 语句
                $countRs = count($rsAll->GetRows());        //统计查询结果
                $countPage = ceil($countRs / $pageSize);    //计算总的页数
                $arrayPageInfo['pageSize'] = $pageSize;     //将每页显示的记录数定义到数组中
                $arrayPageInfo['countRs'] = $countRs;       //将总的记录数定义到数组中
```

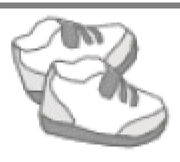


```
                $arrayPageInfo['page'] = $nowPage;          //将当前页码定义到数组中
                $arrayPageInfo['countPage'] = $countPage;  //将总的页码定义到数组中
                $arrayPageInfo['first'] = 1;
                if ($page > 1) {
                    $arrayPageInfo['previous'] = $rs->AbsolutePage() - 1;  //将上一页的数据定义到数组中
                } else {
                    $arrayPageInfo['previous'] = 1;
                }
                if ($page < $countPage) {
                    $arrayPageInfo['next'] = $rs->AbsolutePage() + 1; //将下一页的数据定义到数组中
                } else {
                    $arrayPageInfo['next'] = $countPage;            //剩余页码
                }
                $arrayPageInfo['last'] = $countPage;                //最后的页码
            }
            return $arrayPageInfo;                                  //返回数组
        }
    }
    ?>
```

PageDB 类中定义的 pageData()方法用于返回一个参数数组，该数组包含所有分页信息，例如每页显示的记录数、总页数、总记录数及分页导航链接具体页码值等，这样只需根据返回数组就可以获取所有分页信息，为以后进一步开发作好铺垫。

（3）创建 StaticPage 类动态生成静态页面主要使用技术要点中介绍的 PHP 缓存函数以及字符串函数和文件系统函数。StaticPage 类的代码如下：

```
    class StaticPage{
        function pageBegin (){                                  //指定要生成静态页的开始部分
            ob_start();
        }
        function pageEnd (){                                    //定义要生成静态页的结束部分
            ob_end_flush();
         }
        function getStaticPageName ($url, $extendsName, $defaultPage = 'index'){
                                                                //根据动态地址生成静态页的名称
             $baseName = basename($url);                        //不包含目录的 php 动态页名
                $pageName = substr($baseName, 0, strpos($baseName, '.php'));      //去掉扩展名
                if ($pageName == '') {
                     $pageName = $defaultPage;
                }
                $searchString = substr(strchr($baseName, '?'), 1, strlen(strchr($baseName, '?')));
                                                                //获取查询字符串部分
                $arraySearchString = explode('&', $searchString); //分割查询字符串并保存到数组中
                $searchName = '';
                for ($i = 0; $i < count($arraySearchString); $i ++) {//使用"-"连接查询字符串的值
                     $searchStringTmpValue = strstr($arraySearchString[$i], '=');
                    if ($searchStringTmpValue != '') {
                        $searchName .= '-' . substr($searchStringTmpValue, 1, strlen($searchStringTmpValue));
                    }
                }
                $fileName = $pageName . $searchName . '.' . $extendsName; //生成静态页文件名
                return $fileName;                                 //返回静态页文件名
```




```
        }
        function makeStaicPage ($url, $entendsName, $saveDir){          //生成静态页文件
            $content = ob_get_contents();                  //获取 PHP 页面生成的 HTML 页面内容
            if (! is_dir($saveDir)) {
                mkdir($saveDir);                           //创建保存静态页的目录
            }
            $fileName = $this->getStaticPageName($url, $entendsName);
            $fp = fopen($saveDir . '/' . $fileName, 'w');
            fwrite($fp, $content);                         //将 HTML 内容写入静态页
        }
        function reMakeCondition ($saveDirAndFile, $reMakeTime = -1){
                                                           //设定重新生成静态页的条件
            if ($reMakeTime == - 1) {
                if (file_exists($saveDirAndFile)) {
                    return true;
                }
            } elseif (file_exists($saveDirAndFile)&& time()-filemtime($saveDirAndFile)<$reMakeTime) {
                //如果静态页超出指定的时间范围或不存在，则返回 true，表示将重新生成静态页
                return true;
            }
            return false;
        }
    }
```

在 StaticPage 类中，定义 pageBegin()和 pageEnd()方法分别用来定义所要生成静态页面的开始和结束位置；定义 getStaticPageName()方法根据实际 PHP 页面的请求地址生成静态页面地址；定义 makeStaicPage()方法创建静态页面；定义 reMakeCondition()方法用于指定重新生成静态页面的条件。

StaticPage 类中生成静态页的方法是真正意义地生成静态页面，而非伪静态，其好处是不受服务器局限，如果使用伪静态的方法则要开启 Apache 服务器的 Rewrite 功能。另外，如果用户使用租用的空间，可能有些空间提供商未提供上述技术支持。

（4）创建 index.php 页面，实现数据的分页输出。

首先，在页面最开始调用 StaticPage 类，并根据 reMakeCondition()方法判断与当前 PHP 页面所对应的静态页面是否存在，或者静态页面是否已经过期，如果过期则重新生成，否则使用 header()函数重定向到与该页面所对应的静态页面中，并使用 exit()函数终止该 PHP 页面剩余代码的执行，从而提高代码执行效率。代码如下：

```
<?php
require_once 'StaticPage.php';                         //包含 StaticPage 类
$ststicPage = new StaticPage();                        //对 StaticPage 进行实例化
$ststicPage->pageBegin();                              //定义生成静态页面的起始位置
$dirName = './pages';                                  //定义生成静态页的目录
$fileName = $ststicPage->getStaticPageName($_SERVER['REQUEST_URI'], 'shtml');
                                                       //生成与 PHP 页面对应的静态页名称
if ($ststicPage->reMakeCondition($dirName .'/'. $fileName, 60)){//判断是否需要重新生成静态页
    header('location:pages/' . $fileName); //不需要则重新定向到与该 PHP 页面所对应的静态页面
    exit();
}
$baseUrl = '/TM/Instances/20/17';
require_once 'ConnDB.php';                             //包含数据库连接类
```



```
require_once 'PageDB.php';                          //包含分页类
$connObj = new ConnDB('mysql', 'localhost', 'root', 'root', 'db_database', false); //实例数据库连接类
$connID = $connObj->getConnID();                    //返回数据库连接 ID
?>
```

然后，实例化 PageDB 分页类，通过 for 循环输出数据，并创建分页超链接，完成数据的分页显示。

最后，关闭数据库的连接，调用 SticPage 类的 makeStaicPage()方法生成静态页，调用 StsticPage 类的 pageEnd()方法指定生成静态页面的结束位置。

```
<?php
$connObj->closeConnID();                            //关闭与数据库的连接
$ststicPage->makeStaicPage($_SERVER['REQUEST_URI'], 'shtml', './pages');    //生成静态页面
$ststicPage->pageEnd();                             //指定生成静态页面的结束位置
?>
```

将 PHP 动态页面生成静态页面，其运行结果如图 16.4 所示。



图 16.4　PHP 动态生成静态页

问题 288　如何应用 Zend_mail 发送邮件？

问题阐述

目前很多的网站都支持邮件的发送功能，如注册用户激活、用户密码找回功能等都可以通过邮箱来实现。那么这个邮件的发送功能是如何实现的呢？这就是我们要解决的问题，通过 Zend Framework 框架中的 zend_mail 组件实现邮件的发送功能。

专家解答

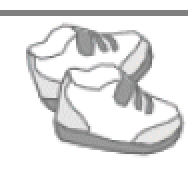
既然是通过 ZendFramework 框架中的组件完成邮件的发送功能，那么首先就要下载 ZendFramework 框架（版本为 ZendFramework-1.11.9）。

然后，将 library 文件夹下的 Zend 目录（组件文件夹）复制到项目指定目录下。

最后，在项目文件中载入发送邮件所需组件（Zend/Mail.php、Zend/Mail/Transport/Smtp.php），调用其中的方法完成邮件的发送操作。其具体方法如下：

（1）通过 include 语句载入所需组件。

（2）获取表单提交的邮件信息，包括发件、收件地址以及邮件标题、内容和附件等信息。



（3）实例化邮件发送类，并且设置其编码格式为 UTF-8。

```
$mail = new Zend_Mail('UTF-8');                    //实例化发送邮件对象
```

（4）定义登录邮件服务器所需的配置信息数组，内容包括邮箱名称、密码、SSL 和端口号。

```
$config = array('auth' => 'login','username' => 'mrsoft8888','password' => 'mrsoft8888','ssl' => 'ssl','port' => '465'); //定义 SMTP 的验证参数
```

（5）实例化邮件登录验证类，其参数为登录邮件服务器的 smtp 地址以及登录邮箱的数据，即上面定义的数组内容。

```
$transport = new Zend_Mail_Transport_Smtp('smtp.sohu.com', $config);    //实例化验证的对象
```

（6）判断表单提交的邮件信息中是否包含附件，如果包含附件，则通过下面的方法获取表单提交的附件信息，并且调用 setMimeboundary()方法定义附件的内容。其关键代码如下：

```
if(isset($_FILES['accessories']['name'])){                    //判断是否包含附件
    $fileName = $_FILES['accessories']['name'];
    $file = $_FILES['accessories']['tmp_name'];
    $fileType = $_FILES['accessories']['type'];
    $contents = file_get_contents($file);                    //PHP 必须大于 4.3 才能使用
    //指定 mime 类型和处理方式 DISPOSITION_ATTACHMENT 附件形式、编码风格、文件名称。其中，fileName 的默认值是 null，我们必须给出
    $mail->createAttachment($contents, $fileType, Zend_Mime::DISPOSITION_ATTACHMENT, Zend_Mime::ENCODING_BASE64, $fileName);
    $mail->setMimeboundary(md5(time() + "any"));
}
```

通过$_FILES[]全局数组获取附件的数据；通过 file_get_contents()将附件中的数据读入到一个字符串中。

通过 createAttachment()方法将上传的附件附加到邮件中。Zend_Mail 会默认认为该文件是二进制对象(application/octet-stream)，以 base64 编码传输，并且作为邮件的附件处理。

通过 setMimeBoundary()方法设置特定的 MIME 分界线。因为在一个包含多个段的邮件中用于分隔邮件不同段的 MIME 分界线(MIME boundary)通常是随机生成的，如果想要通过特定的 MIME 分界线进行控制，那么就需要用到此方法。

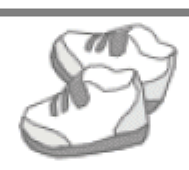
（7）定义邮件的主题、内容和发件人和收件人。

```
$mail->setBodyHtml($mailbody);                         //发送邮件主体
$mail->setFrom($form, 'Zend_Mail 邮件发送测试');        //定义邮件发送邮箱
$mail->addTo($to, 'Zend_Mail 邮件发送测试');            //定义邮件的接收邮箱
$mail->setSubject($subject);                           //定义邮件主题
```

应用 Zend_mail 组件中的基本属性，实现邮件信息的发送，其应用属性如表 16.2 所示。

表 16.2　Zend_Mail 发送邮件的基本属性

值	说　明
addTo($option,$to)	收件人地址，$option为发件人地址，$to为邮件中发件人地址显示
serFrom($option,$from)	发件人地址，$option为收件人地址，$from为邮件中收件人地址显示
setSubject($subject)	邮件标题
setbodyText($body)	邮件内容，文本格式的数据
setbodyHtml($body)	邮件内容，HTML格式的数据



（8）调用 send()方法执行邮件的发送。

```
$mail->send($transport);
?>
```

应用示例

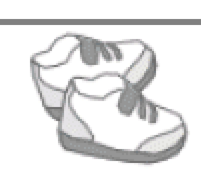
创建一个搜狐邮箱（mrsoft8888@sohu.com，密码为 mrsoft8888），登录搜狐邮箱服务器（smtp.sohu.com），设置其端口号为 465，通过 zend_mail 组件实现邮件的发送功能。其关键代码如下：

```
<?php
include 'Zend/Mail.php';                                    //调用发送邮件的文件
include 'Zend/Mail/Transport/Smtp.php';                     //调用 SMTP 验证文件
$form="mrsoft8888@sohu.com";                                //定义登录使用的邮箱
$to=$_POST['to'];                                           //获取邮件接收地址
$subject=$_POST['subject'];                                 //获取邮件标题
$mailbody=$_POST['content'];                                //获取邮件内容
$mail = new Zend_Mail('UTF-8');                             //实例化发送邮件对象
$config = array('auth' => 'login','username' => 'mrsoft8888','password' => 'mrsoft8888','ssl' => 'ssl',
'port' => '465'); //定义 SMTP 的验证参数
$transport = new Zend_Mail_Transport_Smtp('smtp.sohu.com', $config);     //实例化验证的对象
if(isset($_FILES['accessories']['name'])){                  //判断是否包含附件
    $fileName = $_FILES['accessories']['name'];
    $file = $_FILES['accessories']['tmp_name'];
    $fileType = $_FILES['accessories']['type'];
    $contents = file_get_contents($file);                   //PHP 必须大于 4.3 才能使用
    //指定 MIME 类型和处理方式 DISPOSITION_ATTACHMENT 附件形式、编码风格、文件
名称。其中，fileName 的默认值是 null，我们必须给出
    $mail->createAttachment($contents, $fileType, Zend_Mime::DISPOSITION_ATTACHMENT,
Zend_Mime::ENCODING_BASE64, $fileName);
    $mail->setMimeboundary(md5(time() + "any"));
}
$mail->setBodyHtml($mailbody);                              //发送邮件主体
$mail->setFrom($form, 'Zend_Mail 邮件发送测试');             //定义邮件发送邮箱
$mail->addTo($to, 'Zend_Mail 邮件发送测试');                 //定义邮件的接收邮箱
$mail->setSubject($subject);                                //定义邮件主题
$mail->send($transport);
?>
```

运行结果如图 16.5 所示。



图 16.5　邮件发送



关于示例的完整代码可参考本书配套资源，由于篇幅所限，这里不再赘述。

专家点评



在通过 Zend_mail 组件发送邮件时，除了项目本身所应用的各种技术方法外，还有一点必须注意，因为它登录的是一个指定的邮箱，并且通过 SMTP 对登录的邮箱进行验证，只有登录成功后才能通过该邮箱向其他邮箱发送邮件，所以我们必须先注册一个邮箱，然后确定邮箱对应的 SMTP 服务器，最终确定邮件登录配置数据，应用 zend_mail 组件中方法实现邮件发送功能。

问题 289　如何解决使用 mail()函数发送邮件时的标题乱码问题？

问题阐述

PHP 程序使用 mail()函数发送邮件时，在接收到的邮件的标题中的中文字符串经常会出现乱码。如何解决这个问题呢？

专家解答

在接收到的邮件标题中，中文字符串出现乱码，追根溯源是在发送邮件时没有应用 base64_encode()函数对字符串进行编码导致的。

base64_encode()函数，应用 base64 对 data 进行编码。编码后的数据要比原始数据多占用 1/3 左右的空间。函数语法如下：

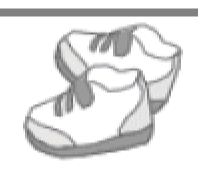
```
string base64_encode( string data )
```

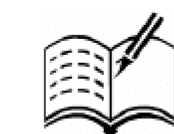
应用 base64 对 data 进行编码。应用 base64 编码后的字符串只包含英文字母大小写、阿拉伯数字、加号与反斜线，共 64 个基本字符，不包含其他的特殊字符，因此，被命名为 base64。

通过 mail()函数发送邮件的正确方法如下：

```
$mailfrom=$_POST['mailfrom'];                              //获取发件人地址
$mailto=$_POST['mailto'];                                  //获取收件人地址
$mailcc=$_POST['mailcc'];                                  //获取抄送人地址
$mailbcc=$_POST['mailbcc'];                                //获取密送人地址
$mailsubject=base64_encode($_POST['mailsubject']);         //获取邮件主题
$mailbody = base64_encode($_POST['mailbody']);             //获取邮件内容
$headers = "From:$mailfrom \r\n Cc:$mailcc \r\n Bcc:$mailbcc";   //定义邮件头
$headers .= "Reply-To:$mailfrom\r\n";
$headers .= "Content-type: text/html;\r\n charset=iso-8859-1 \r\n";
if(@mail($mailto, $mailsubject,$mailbody,$headers)){//通过 mail()函数发送邮件，并给出邮件发送结果
    echo "<script>alert('邮件发送成功！ ');history.back()</script>";
}else{
    echo "<script>alert('邮件发送失败！ ');history.back()</script>";
}
```

在应用了 base64_encode()函数对标题和内容进行编码后，输出的中文字符串就不会出现乱码了。既然对输出的字符串进行了编码，那么在输出时就需要应用对应的函数对其进行解码，





这样才能保证数据的正常输出，其解码应用的是 base64_decode()函数。

base64_decode()函数对使用 MIME base64 编码的数据进行解码，并返回原始数据，返回的数据可能是中文字符串或二进制数据，失败则返回 false。函数语法如下：

```
string base64_decode ( string encoded_data )
```

问题 290　如何实现带附件的邮件发送？

问题阐述

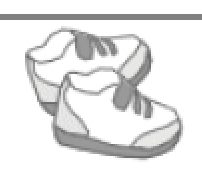
在实现邮件发送功能的同时，也可以实现附件的发送功能。通过附件可以实现更多有价值信息的传递。那么这个附件的发送功能是如何实现的呢？下面我们就来探讨一下这个问题。

专家解答

发送带附件的邮件并不是简单地将附件内容上传到 POP3 服务器的某个目录下，而是首先将附件进行二进制编码，之后将该编码连接到邮件内容之后，当然邮件内容和附件编码之间需要用标记进行分割，该标记在邮件头中定义。下面内容为带附件的邮件组织形式：

```
Received: from mrkj ([192.168.1.227]) by MRZTS with Microsoft SMTPSVC(6.0.3790.0);
Mon, 18 Dec 2006 09:02:40 +0800
Date: Mon, 18 Dec 2006 00:57:26 +0000
Subject: 12
To: zts***@163.com
Content-type: multipart/mixed; Boundary= 4585e7760018e
From:zts***@163.com
Return-Path: zts***@163.com
Message-ID:
X-OriginalArrivalTime: 18 Dec 2006 01:02:40.0187 (UTC) FILETIME=[372680B0:01C72240]
--4585e7760018
Content-type: text/plain; charset=iso-8859-1
Content-transfer-encoding: 8bit
 (此处为邮件内容)
--4585e7760018
Content-type: text/plain; name=shop.txt
Content-disposition: attachment; filename=shop.txt
Content-transfer-encoding: base64
wffV4rXEtcS1xLXEtcS1xLTvtb21xNK7ICC13LXctcS1xA== （此处为附件的 base64 编码）
--4585e7760018--
```

带附件的电子邮件的不同部分之间用分界线来分隔，分界线在 Content--type 头中定义，例如“Boundary= 4585e7760018e”，其中“4585e7760018e”表示不同部分的分界线。邮件的每个新部分以两个连字号（--）和分界线开始，最后一个分界线后也有两个连字号，表示这个邮件中没有其他部分了。在每个分界线后有一些行，用来告诉邮件程序这部分内容的类型，如上面例子中第一个分界线后面的以 Content-type: text/plain 开头的行，这些行说明后面的部分是 ISO-8859-1 字符集的纯文本形式，跟在第二个分界线后的行告诉邮件程序现在的部分是一个 txt 文件，它的名字是“shop.txt”。Content-disposition 行告诉邮件程序如果可能就以内嵌的方式显示附件，现在新的邮件程序会在消息后显示 txt 文件的内容，如果 Content-disposition 被设为



attachment，那么邮件程序就不会显示 txt 文件的内容，而是显示一个连接到文件的图标，收件人要看附件的内容，必须单击这个图标。一般情况下，如果附件是一些文本，则 Content-disposition 就会被设为 inline，这是因为现在大部分邮件程序能够不借助其他浏览器而直接显示附件（文本）的内容。如果附件不是文本（如图片或其他类似的内容），Content-disposition 就设为 attachment。



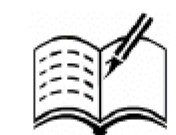
Note

应用示例

应用 mail()函数实现邮件的发送功能，并且支持附件的发送。其关键代码如下：

```
<!--*****************************sendmail.php***********************************-->
<?php
$mailfrom=$_POST['mailfrom'];                                    //获取发件人地址
$mailto=$_POST['mailto'];                                        //获取收件人地址
$mailcc=$_POST['mailcc'];                                        //获取抄送人地址
$mailbcc=$_POST['mailbcc'];                                      //获取密送人地址
$mailsubject=base64_encode($_POST['mailsubject']);               //获取邮件主题
$mailbody = base64_encode($_POST['mailbody']);                   //获取邮件内容
if(empty($_FILES['upload_file']['name'])){
    $headers = "From:$mailfrom \r\n Cc:$mailcc \r\n Bcc:$mailbcc"; //定义邮件头
    $headers .= "Reply-To:$mailfrom\r\n";
    $headers .= "Content-type: text/html;\r\n charset=iso-8859-1 \r\n";
    if(@mail($mailto, $mailsubject,$mailbody,$headers)){         //通过 mail()函数发送邮件，
                                                                  并给出邮件发送结果
          echo "<script>alert('邮件发送成功！ ');history.back()</script>";
    }else{
          echo "<script>alert('邮件发送失败！ ');history.back()</script>";
    }
}else{
    $file = $_FILES['upload_file']['tmp_name'];                  //获取附件名称
    $boundary = uniqid("");                                      //定义分界线
    $headers = "MIME-Version: 1.0 \n";
    $headers .= "Content-type: multipart/mixed; Boundary= $boundary \r\n";
    $headers .= "From:$mailfrom\r\n";                            //定义邮件头
    if($_FILES['upload_file']['type']){                          //判断附件类型
         $mimeType = $_FILES['upload_file']['type'];
     }else{
         $mimeType ="application/unknown";                       //获取附件类型
     }
    $fileName = $_FILES['upload_file']['name'];                  //获得附件名称
    $fp = fopen($file, "r");                    //打开文件
    $read = fread($fp, filesize($file));        //将附件读入变量$read
    $read = base64_encode($read);               //转换 base64 编码
    $read = chunk_split($read);                 //将长字符串切成由每行 76 个字符组成的小块
    /*定义邮件体*/
    $body = "--$boundary--
Content-type: text/plain; charset=iso-8859-1
$mailbody
--$boundary--
Content-type: $mimeType; name=$fileName
Content-disposition: attachment; filename=$fileName
```





```
Content-transfer-encoding: base64
$read
--$boundary--";
    if(mail($mailto, $mailsubject,$body,$headers)){                    //用 mail()函数发送邮件
        echo "<script>alert('附件发送成功！');history.back()</script>";
    }else{
        echo "<script>alert('附件发送失败！');history.back()</script>";
    }
}
?>
```

专家点评

当所发送的邮件中带有附件时，首先需要将邮件的 MIME 的 Content-type 类型定义成 multipart/mixed，然后定义邮件内容分割线（一串字符或数字），最后将文件内容编写在邮件主体内容中进行发送。

注意，PHP 的 mail()函数只能将邮件发送到 SMTP 的邮件转发器中，然后通过转发器将邮件发送出去。

问题 291　如何实现邮件群发？

问题阐述

人们在工作中经常会将相同内容邮件发送给多个用户，如果一一发送，将会给工作带来很大的麻烦，也许还会带来经济上的损失，为了使工作更加快捷、方便和降低出错率，可使用邮件群发功能。那么如何才能实现邮件群发呢？

专家解答

要实现邮件群发主要应用 mail()函数，原理是通过循环将邮件一一发送，关键是如何解读以“|”分隔的多个“收件人”邮箱。其主要应用函数 explode()。explode()函数的语法如下：

```
array explode ( string separator, string string [, int limit])
```

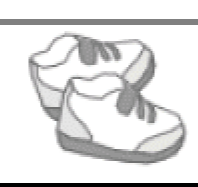
explode()的参数说明如表 16.3 所示。

表 16.3　explode()的参数说明

参　　数	性　　质	说　　明
separator	必选	规定以什么字符串分割字符串
string	必选	要分割的字符串
limit	可选	规定所返回的数组元素最大数量

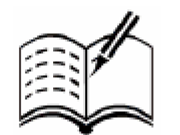
说明：

若参数 limit 被设置，则返回数组中最多包含 limit 个元素，而最后那个元素将包含参数 String 的剩余部分；如果 limit 为负数，则返回除了最后-limit 个元素外的所有元素，已是 PHP 5.1.0 中新增的特性，用户使用时应注意 PHP 版本。



注意：

参数 limit 是 PHP 4.0.1 中新增的。由于历史原因，虽然 implode()函数可以接收两种参数顺序，但是 explode()不行，必须参数 Separator 在参数 String 之前。



Note

应用示例

下面通过一个具体的示例来讲解邮件群发的实现方法。其具体实现过程如下：

（1）创建 index.php 文件，完成邮件群发页面的设计。

（2）在 index.php 页面中根据提交按钮的值判断用户是否提交表单。通过 explode()函数将获取的收件人以“|”分割为 N 个元素，以 for 循环将得到的 N 个元素一一通过 mail()函数发送。其关键代码如下：

```
if($_POST[Submit]!=""){
    //开始得到 mail 函数中的$to 分隔为 N 个
    $tomany=explode("|",$to);
    //然后做一个简单的判断：如果只发送了一个邮件那么直接发送
    for($i=0;$i<count($mailnum);$i++){
        if(@mail($tomany[$i],$title,$content,$headers)){
            echo"<script>alert('邮件发送成功！');</script>";
        }else{
            echo"<script>alert('邮件发送失败！');</script>";
        }
    }
}
```

邮件群发的运行结果如图 16.6 所示，将多个邮箱间以“|”分隔，同时将同一邮件发送给多个用户。



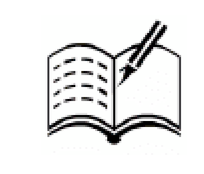
图 16.6　邮件群发

问题 292　自定义在线编辑器的制作

问题阐述

在线编辑器的功能经常会出现在论坛、博客或微博中，其作用是让用户发布的内容更加丰富多彩。但是，不知道大家是否注意到，在运行一些带有在线编辑器功能的页面时，在线编辑器功能的加载有时会很慢，网速固然是一个重要的原因，但是由于在线编辑器功能的强大，导





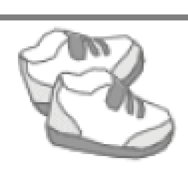
致加载缓慢，这个原因也不能忽视。那么，我们是否可以模拟目前比较流行的 FCKediter 在线编辑器，编写一个属于自己的在线编辑器呢？这样我们就可以根据需求对这个编辑器进行量身定做，既可以满足我们的需求，又能够对其进行精简，让它在加载过程中更加清晰、流畅。

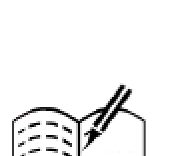
专家解答

自定义在线编辑器其实就是目前已经成熟的在线编辑器的精简版，其应用 jQuery 技术，将操作方法封装到自定义函数中，最终存储于 LzhEditor.js 文件中。其封装的部分方法如下：

```
/**
 * 执行命令
 */
function lzhEditorFormat(hc, pa) {
    window.frames["editor"].focus();
    window.frames["editor"].document.selection.createRange();
    if (pa == "") {
        window.frames["editor"].document.execCommand(hc, false);
    } else {
        window.frames["editor"].document.execCommand(hc, false, pa);
    }
}
/**
 * 创建链接
 */
function lzhEditorCreateLink() {
    var linkStr = prompt("\u8bf7\u8f93\u5165\u94fe\u63a5URL\u5730\u5740\uff01",
            "");
    if (linkStr != null) {
        lzhEditorFormat("CreateLink", linkStr);
    }
}

/**
 * 显示添加表情层
 */
function lzhEditorAddFace() {
    $("#faceLayer").css("display", "block");
}
/**
 * 插入表情
 */
function insertFace(faceUrl) {
    if (faceUrl != null && faceUrl != "") {
        lzhEditorFormat("InsertImage", faceUrl);
    }
    $("#faceLayer").css("display", "none");
}
/**
 * URL 方式添加图片
 */
function lzhEditorInsertImage() {
    if ($.trim($("#imageUrl").val()) == "") {
        alert("请输入要插入图片的 URL 地址！");
```






```
        } else {
            var v = $.trim($("#imageUrl").val());
            var extsName = v.substring(v.lastIndexOf(".") + 1).toLowerCase();
            if (!(extsName == "gif" || extsName == "jpg" || extsName == "png" || extsName == "bmp")) {
                alert("插入的图片只能为 gif、jpg、png 或 bmp 格式！");
            } else {
                lzhEditorFormat("InsertImage", $.trim($("#imageUrl").val()));
                $("#uploadLayer").css("display", "none");
            }
        }
    }
    /**
     * 显示插入图片层
     */
    function lzhEditorAddImage() {
        $("#uploadLayer").css("display", "block");
    }
    /**
     * 更改编辑器高度
     */
    function changeEditorArea(type) {
        if (type == 1) {
            document.getElementById("editor").height = parseInt(document
                    .getElementById("editor").height) +10;
        } else {
            if (parseInt(document.getElementById("editor").height) >=50) {
                document.getElementById("editor").height = parseInt(document
                        .getElementById("editor").height) - 10;
            }
        }
    }
    /**
     * 更改插入图片方式层
     */
    function changUploadLayer(x) {
        if (x == 0) {
            $("#upLayer1").css("display", "block");
            $("#upLayer2").css("display", "none");
        } else {
            $("#upLayer1").css("display", "none");
            $("#upLayer2").css("display", "block");
        }
    }
    /**
     * 将编辑器内容赋给隐藏域（表单提交时调用）
     */
    LzhEditor.prototype.Submit = function() {
        var content = window.frames["editor"].document.body.outerHTML;
        content = content
                .replace(
                        '<BODY style="PADDING-RIGHT: 5px; PADDING-LEFT: 5px; FONT-SIZE:
12px; PADDING-BOTTOM: 5px; MARGIN: 0px; PADDING-TOP: 5px">',
                        '');
```



```
            content = content.replace('</BODY>', '');
            document.getElementById(this.fieldName).value = content;
        };
```

在完成脚本的封装之后，就需要在网页中载入脚本文件，其前提是必须在网页中载入 jQuery 组件，否则不能调用脚本文件中的方法。

接着，编写 JavaScript 脚本，实例化自定义在线编辑器类，传递参数（包括名称、大小、脚本文件存储位置等），完成网页中自定义在线编辑器的输出，其关键代码如下：

```
    <script language="javascript">
        var lzhEditor = new LzhEditor('content','500','200','','js/LzhEditor');
        lzhEditor.Create();
    </script>
```

最后，通过$_POST[]获取在线编辑器中提交的数据，切记必须要应用 stripslashes()函数对在线编辑器传递的数据进行解析，否则内容将无法正常输出。

应用示例

应用自定义在线编辑器编辑内容，并且获取自定义在线编辑器提交的值，其运行结果如图 16.7 所示。

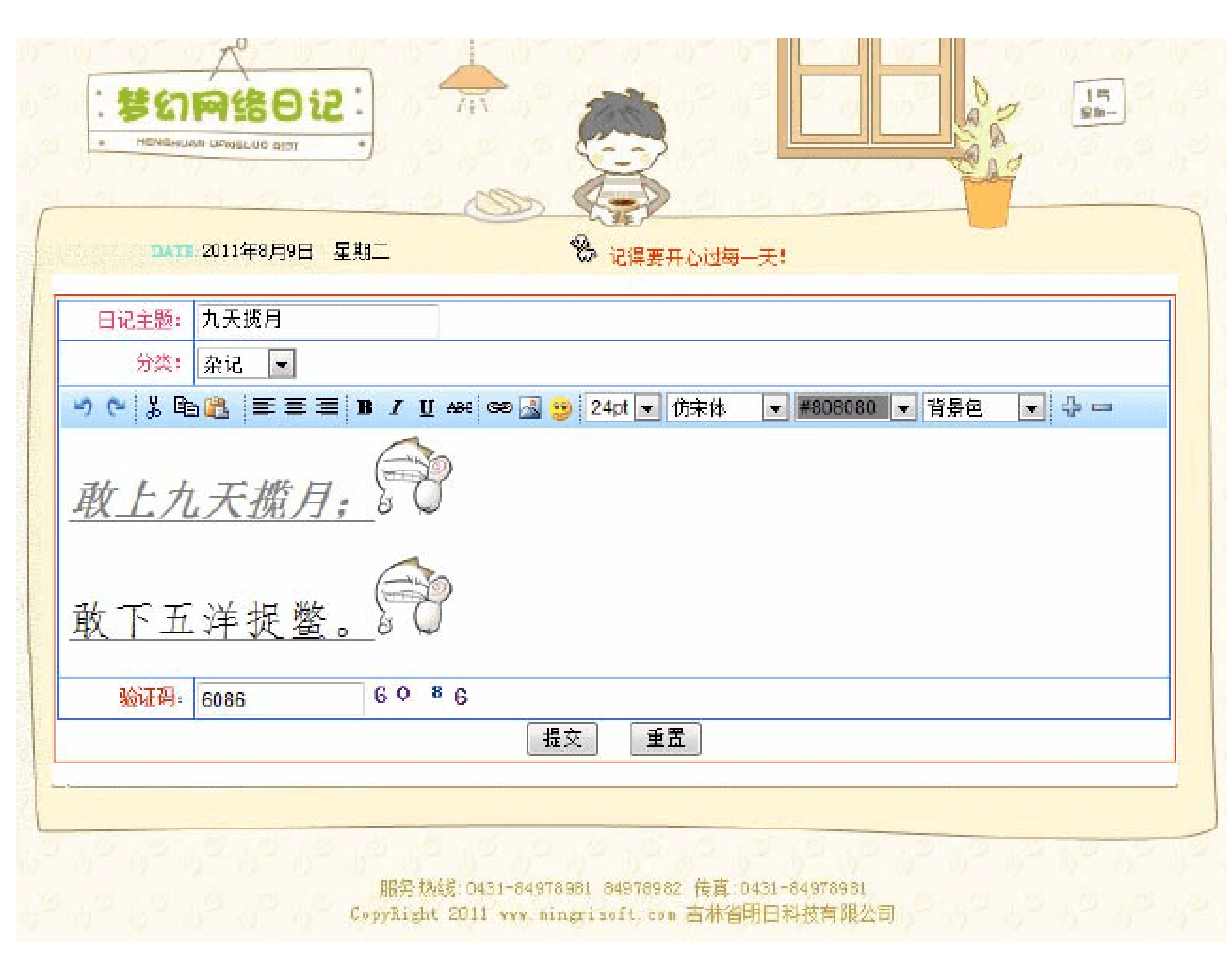


图 16.7　自定义在线编辑器应用

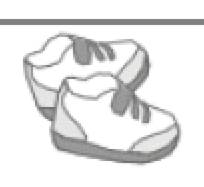
专家点评

应用自定义在线编辑器，其首先要在网页中载入 jQuery.js 和 LzhEditor.js 文件，然后在网页中正确地调用脚本中的方法，传递合适的参数，最后在获取在线编辑器提交的值时应用 stripslashes()函数对其进行解析。

问题 293　如何应用 CKEditor 网页编辑器？

问题阐述

论坛、博客、微博或在线视频类网站都具备发表评论的功能，为了使评论信息更加醒目，



可以通过在线编辑器对评论的内容进行在线编辑。那么如何应用在线编辑器呢？

专家解答



在这里向读者介绍 CKEditor 网页编辑器的使用方法。首先下载和安装 CKEditor 编辑器。

CKEditor 是开源网页编辑软件，FCKEditor 在 2009 年发布更新的 3.0 版本。CKEditor 是 FCKeditor 的一个完全重写版本，它弥补了 FCKeditor 打开速度慢的缺陷，并提供了强大的、可扩展的 JavaScript API。

说明：

原来叫 FCK，是因为最初的开发者叫 Frederico Calderia Knabben; 现在叫 CK，意指“Content and Knowledge”。

CKEditor 的安装步骤非常简单，如下所示：

（1）从 CKEditor 官方网站上下载 CKEditor 的最新版本。

（2）将下载的文件解压到项目根目录下的 ckeditor 文件夹。

安装成功后，可以通过编辑器附带的一些实例来验证其是否能够正常运行。附带实例存储于“_samples“目录下。访问地址是：http://<网站域名>/<CKEditor 安装路径>/_samples/index.html。

CKEditor 安装成功后，就需要在网页中嵌入编辑器了。在网页中嵌入 CKEditor 编辑器的方法有很多，在这里使用 PHP 脚本将 CKEditor 嵌入到网页中，代码如下：

```
<?php
include("ckeditor/ckeditor.php");                                    //包含 ckeditor 类
$CKEditor = new CKEditor();                                          //类的实例化
$CKEditor->basePath = 'ckeditor/';                                   //设置 ckeditor 的目录
$CKEditor->config['width'] = 600;                                    //设置宽
$CKEditor->textareaAttributes = array("cols" => 20, "rows" => 10);   //设置文本的行、列
$CKEditor->replaceAll();
?>
```

应用示例

通过 CKEditor 对博客中发表文章的内容进行编辑，包括对文字的处理、添加表格、上传图片和上传 Flash 等。其具体实现过程如下：

（1）下载、安装 CKEditor 编辑器。

（2）创建 index.php 文件，创建表单，通过 PHP 脚本嵌入 CKEditor 编辑器，将博客文章提交到 index_ok.php 文件中。其关键代码如下：

```
<form name="form1" method="post" action="index_ok.php" onSubmit="return chkinput(this)">
    <input name="topic" type="text" value="" size="40" />
    <input name="author" type="text" value="" size="30" />
    <input name="comefrom" type="text" value="" size="40" />
    <input type="hidden" name="langver" value="ch" />
    <textarea cols="80" id="content" name="content" rows="10">请输入文章内容;</textarea>
    <input type="image" name="imageField2" src="images/bg_04.jpg" />
    <input type="image" name="imageField" src="images/bg_06.jpg" onClick="form_reg.reset()" style="cursor:hand" />
</form>
<?php
include("ckeditor/ckeditor.php");                                    //包含 ckeditor 类
```






```
    $CKEditor = new CKEditor();                                          //类的实例化
    $CKEditor->basePath = 'ckeditor/';                                   //设置 ckeditor 的目录
    $CKEditor->config['width'] = 600;                                    //设置宽
    $CKEditor->textareaAttributes = array("cols" => 20, "rows" => 10);   //设置文本的行、列
    $CKEditor->replaceAll();
?>
```

运行结果如图 16.8 所示。

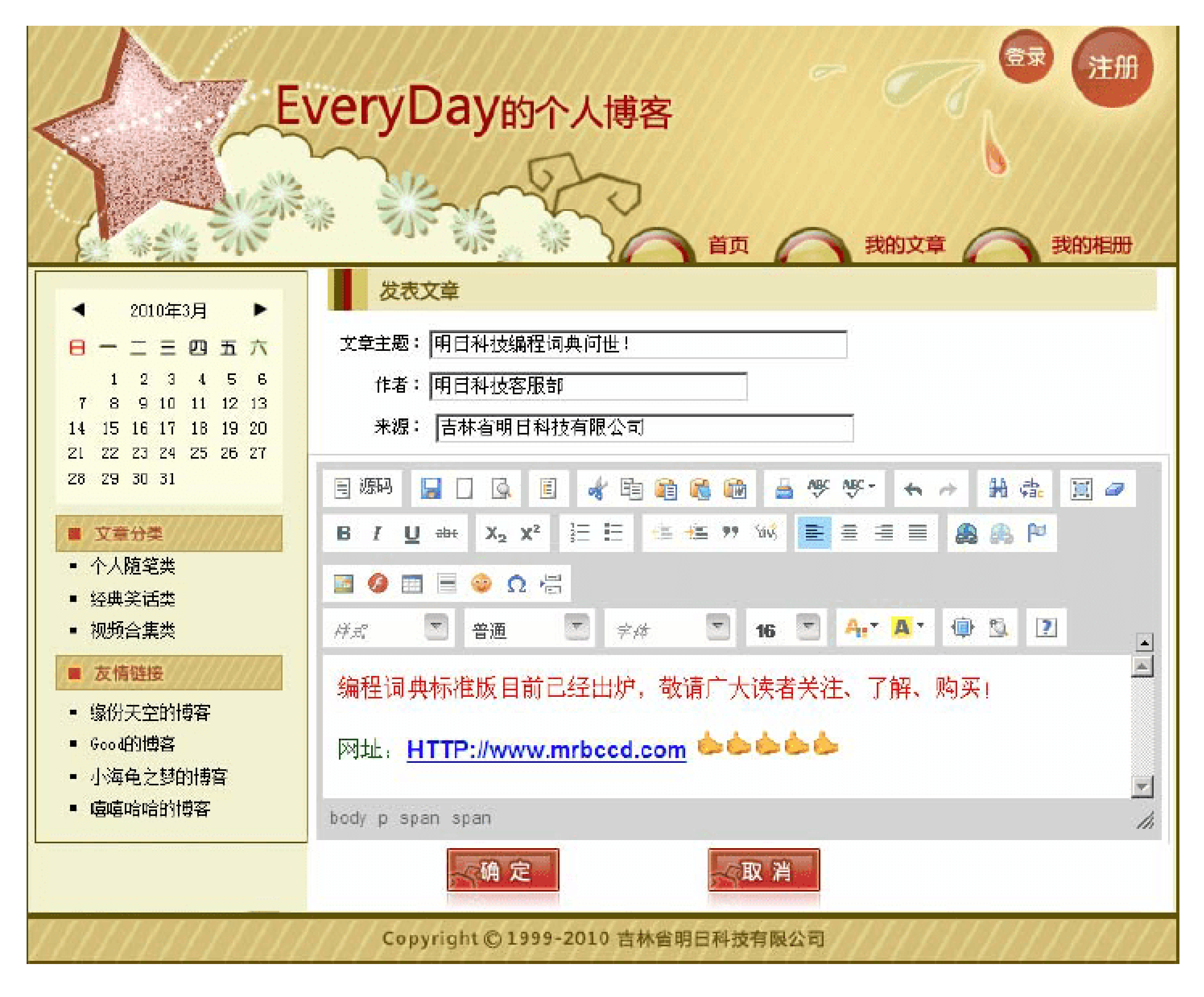


图 16.8　CKEditor 的应用

（3）在 JavaScript 脚本中创建 chkinput()方法，验证提交的数据是否为空。

（4）创建 system 文件夹，载入 ADODB 类库，编写连接、操作数据库类的文件 system.class.inc.php 和实例化类的文件 system.inc.php。

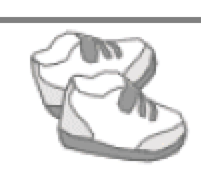
（5）创建 index_ok.php 文件，获取表单中提交的数据，然后通过数据库操作类中的方法将数据添加到数据表中。

专家点评

在网页中嵌入 CKEditor 网页编辑器时，一定要设置正确的路径，如果路径设置得不正确，那么将弹出如图 16.9 所示的错误信息。



图 16.9　路径设置不正确导致的错误



之所以会出现上面的错误，是因为没有找到 CKEditor，其错误的代码如下：

```
<?php
include("ckeditor/ckeditor.php");                                    //包含 ckeditor 类
$CKEditor = new CKEditor();                                          //类的实例化
$CKEditor->basePath = '/ckeditor/';                                  //设置 ckeditor 的目录
$CKEditor->config['width'] = 600;                                    //设置宽
$CKEditor->textareaAttributes = array("cols" => 20, "rows" => 10);   //设置文本的行、列
$CKEditor->replaceAll();
?>
```



Note

上述问题就出在$CKEditor->basePath 的值上，如果 CKEditor 存储在根目录下的 ckeditor 文件夹下，那么$CKEditor->basePath 的值应该是“ckeditor/”。

问题 294　如何利用 WebBrowser 打印报表？

问题阐述

WebBrowser 是 IE 内置的浏览器控件，无须用户下载。它的优点是客户端独立完成打印目标文档的生成，减轻服务器负荷；缺点是源文档的分析操作复杂，并且要对源文档中要打印的内容进行约束。那么，如何利用 WebBrowser 打印报表呢？

专家解答

IE 内置的 WebBrowser 控件实现报表的打印等功能，该控件的具体参数如下：

document.all.WebBrowser.Execwb(7,1)：表示打印预览。

document.all.WebBrowser.Execwb(6,1)：表示打印。

document.all.WebBrowser.Execwb(6,6)：表示直接打印。

document.all.WebBrowser.Execwb(8,1)：表示页面设置。

document.all.WebBrowser.Execwb(1,1)：打开页面。

document.all.WebBrowser.Execwb(2,1)：关闭所有打开的 IE 窗口。

document.all.WebBrowser.Execwb(4,1)：保存网页。

document.all.WebBrowser.Execwb(10,1)：查看页面属性。

document.all.WebBrowser.Execwb(17,1)：全选。

document.all.WebBrowser.Execwb(22,1)：刷新。

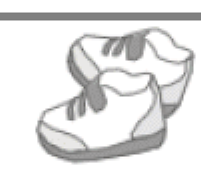
document.all.WebBrowser.Execwb(45,1)：关闭窗体无提示。

通过 HTML 的 object 标签，调用 WebBrowser 控件，代码如下：

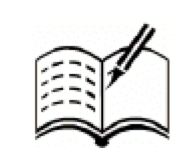
```
<object id="WebBrowser" classid="ClSID:8856F961-340A-11D0-A96B-00C04Fd705A2" width="0" height="0">
</object>
```

调用 WebBrowser 控件的相应参数实现打印预览、打印等功能，代码如下：

```
<a href="#" onClick="document.all.WebBrowser.Execwb(7,1)">打印预览</a>
<a href="#" onClick="document.all.WebBrowser.Execwb(6,1)">打印</a>
<a href="#" onClick="document.all.WebBrowser.Execwb(6,6)">直接打印</a>
<a href="#" onClick="document.all.WebBrowser.Execwb(8,1)">页面设置</a>
```

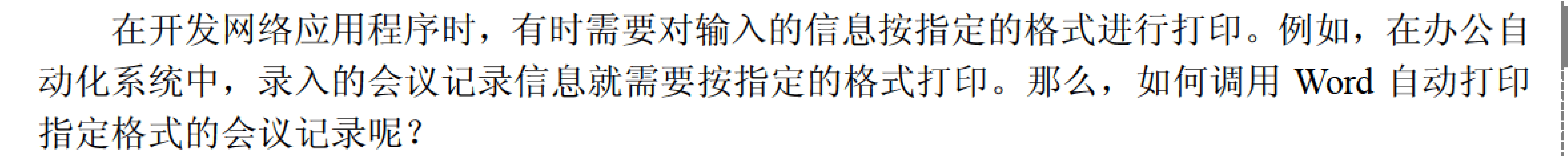


问题 295　如何调用 Word 自动打印指定格式的会议记录？



问题阐述

在开发网络应用程序时，有时需要对输入的信息按指定的格式进行打印。例如，在办公自动化系统中，录入的会议记录信息就需要按指定的格式打印。那么，如何调用 Word 自动打印指定格式的会议记录呢？



专家解答

在 PHP 中利用 Word 自动打印指定格式的会议记录可以总结为以下几点：

（1）应用 JavaScript 的 ActiveXObject()构造函数创建一个 Word Application 对象的实例。

（2）打开指定的 Word 文档，这里需要应用 PHP 的函数获取模板文档所在路径，代码如下：

```
$path="http://".$HTTP_HOST.dirname($PHP_SELF)."/meetingDot.doc";
```

其中，$HTTP_HOST 变量用于获取服务器名称；dirname($PHP_SELF)用于获取路径中的路径名。

（3）通过 Word Application 对象的 Bookmarks 集合的相应方法将表单内容写入指定的 Word 文档中。

（4）调用 wdapp.Application.Printout();实现自动打印 Word 文档。

应用示例

在 PHP 中调用 Word 自动打印指定格式的会议记录。其关键步骤如下：

（1）创建一个 Word 文档，在该文档中设计好要打印的会议记录的格式，并将其保存到示例根目录下，名称为 meetingDot.doc。

（2）在创建好的 Word 文档中的指定位置插入书签。插入书签的方法如下：首先选中需要替换的文本，然后选择“插入”→“书签”命令，在打开的对话框中输入书签名，并单击“添加”按钮即可。

（3）在示例主页面中添加用于收集会议信息的表单及表单元素。

（4）应用 PHP 获取模板文档所在的路径，代码如下：

```
<?php
//获取模板文档所在的路径
$path="http://".$HTTP_HOST.dirname($PHP_SELF)."/meetingDot.doc";
?>
```

（5）编写自定义 JavaScript 函数 outDoc()，用于将表单收集的数据导出到 Word，并进行自动打印。代码如下：

```
<?php
//定义新型字符串
$str=<<<word
<script language="javascript">
function outDoc(){
	var wdapp=new ActiveXObject("Word.Application");
	wdapp.visible=true;
```



Note

```
        wddoc=wdapp.Documents.Open("$path");            //打开指定的文档
        var form=document.all.form1;
        title=form.title.value;
        meetingTime=form.meetingTime.value;
        compere=form.compere.value;
        attend=form.attend.value;
        content=form.content.value;
        //输出会议标题
        range =wdapp.ActiveDocument.Bookmarks("title").Range;
        range.Text=title;
        //输出会议时间
        range =wdapp.ActiveDocument.Bookmarks("meetingTime").Range;
        range.Text=meetingTime;
        //输出会议主持人
        range =wdapp.ActiveDocument.Bookmarks("compere").Range;
        range.Text=compere;
        //输出出席人员
        range =wdapp.ActiveDocument.Bookmarks("attend").Range;
        range.Text=attend;
        //输出会议内容
        range =wdapp.ActiveDocument.Bookmarks("content").Range;
        range.Text=content;
        wddoc.Application.Printout();
        wdapp=null;
    }
    </script>
    word;
    echo $str;
    ?>
```

（6）通过单击“Word 打印”按钮调用自定义 JavaScript 函数 outDoc()。关键代码如下：

```
<input name="Submit" type="button" class="btn_grey" onClick="outDoc()" value="Word 打印">
```

运行本示例，在页面中输入相应的会议信息，单击“Word 打印”按钮，即可将录入的会议信息导出到指定的 Word 文档中，并自动按该文档指定的格式打印。运行结果如图 16.10 所示。

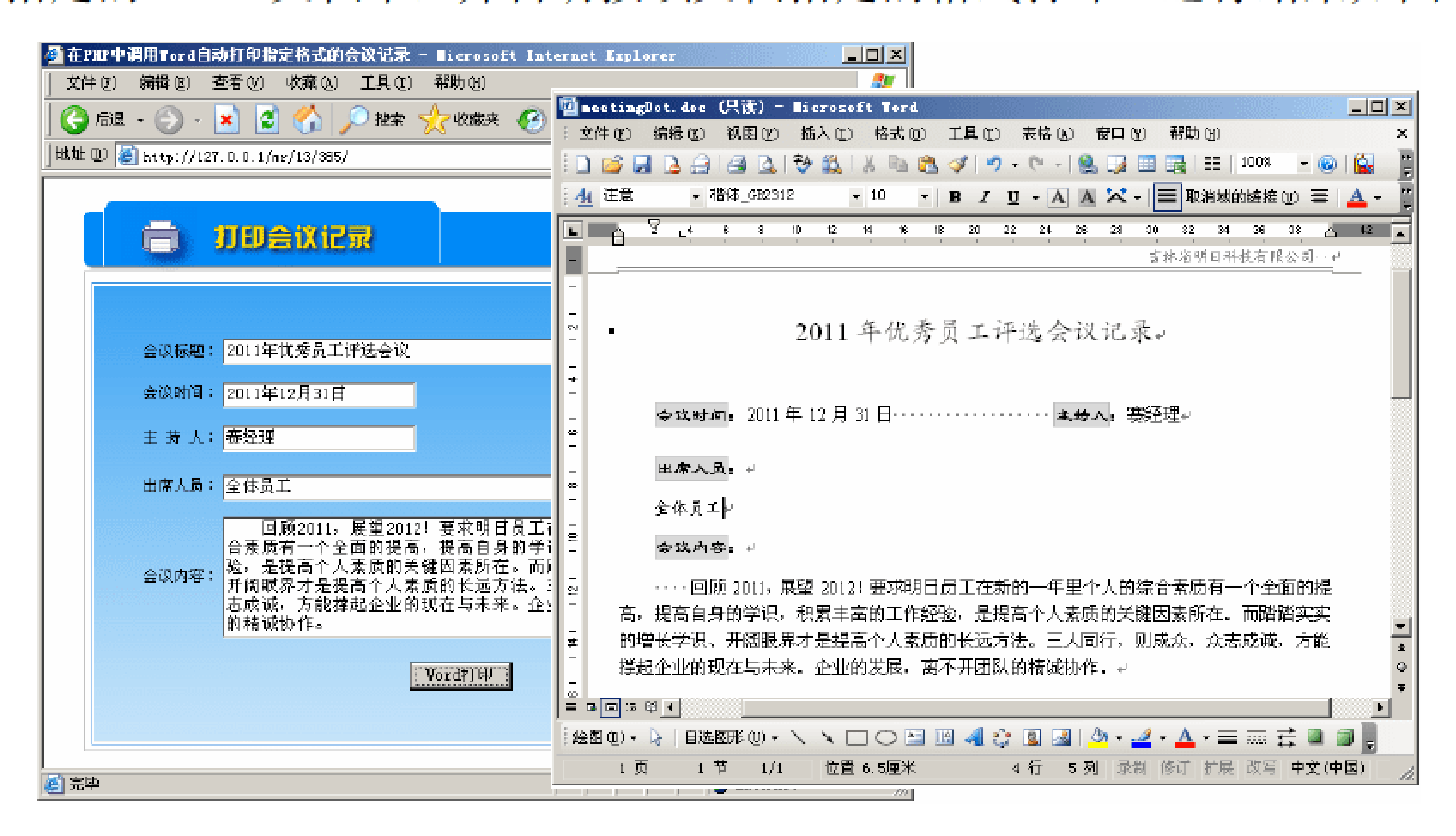


图 16.10　在 PHP 中调用 Word 自动打印指定格式的会议记录





问题 296　如何将 Web 页面中的数据导出到 Excel 中并自动打印?

问题阐述

在 Web 网站的实际项目开发过程中，经常需要将 Web 页面中的数据导出到 Excel 中并实现自动打印。那么，如何实现这一操作呢?

专家解答

通过 PrintOut()方法可实现自动打印 Excel 工作表中的内容。该方法用于打印指定对象。其语法格式如下：

```
expression.PrintOut(From, To, Copies, Preview, ActivePrinter, PrintToFile, Collate, PrToFileName)
```

参数说明：

❶ expression：必选参数。用于返回“Chart 对象”、“Charts 集合对象”、“Range 对象”、“Sheets 集合对象”、“Window 对象”、“Workbook 对象”、“Worksheet 对象”或“Worksheets 集合对象”中的某个对象。

❷ From：可选参数。用于指定打印的开始页号。如果省略该参数，将从起始位置开始打印。

❸ To：可选参数。用于指定打印的终止页号。如果省略该参数，将打印至最后一页。

❹ Copies：可选参数。用于指定要打印的份数。如果省略该参数，将只打印一份。

❺ Preview：可选参数。值为 true 或 false，如果为 true，则 Microsoft Excel 打印指定对象之前进行打印预览；如果为 false 或者省略此参数，则立即打印该对象。

❻ ActivePrinter：可选参数。用于设置活动打印机的名称。

❼ PrintToFile：可选参数。值为 true 或 false，如果为 true，则打印输出到文件。如果没有指定 PrToFileName，则 Microsoft Excel 将提示用户输入要输出文件的文件名。

❽ Collate：可选参数。当值为 true 时，逐份打印每份副本。

❾ PrToFileName：可选参数。如果将 PrintToFile 设置为 true，则本参数指定要打印到的文件名。

说明：

From 参数和 To 参数所描述的“页”指的是要打印的页，并非指定工作表或工作簿中的全部页。

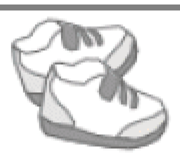
应用示例

将 Web 页面中的话费列表导出到 Excel 中，同时实现保存并自动打印功能。其具体实现步骤如下：

（1）应用 include 命令引用数据库配置文件来访问数据源，代码如下：

```
<?php include "conn/conn.php"; ?>
```

（2）应用 do…while 循环语句以表格形式输出话费信息到浏览器中，并将显示话费信息的表格的 id 设置为 cost，因为要将该表格中的数据导出到 Excel 中。关键代码如下：





```
<?php
    $sql=mysql_query("select * from tb_telcost order by month desc");
    $info=mysql_fetch_array($sql);
    do{
 ?>
  <tr align="center" bgcolor="#FFFFFF">
        <td width="14%" height="30"><?php echo $info[tel];?></td>
        <td width="13%"><?php echo $info[bdcost];?></td>
        <td width="14%"><?php echo $info[ipcost];?></td>
        <td width="14%"><?php echo $info[ldcost];?></td>
        <td width="14%"><?php echo $info[ldcost];?></td>
        <td width="16%"><?php echo $info[total];?></td>
        <td width="15%"><?php echo $info[month];?></td>
    </tr>
  <?php
      }while($info=mysql_fetch_array($sql));
  ?>
```

（3）编写自定义 JavaScript 函数 outExcel()，用于将 Web 页面中的话费列表信息和自定义的标题信息导出到 Excel 中，并保存到 Excel 的默认工作目录中，同时自动调用打印机进行打印并实现自动打印。代码如下：

```
<script language="javascript">
function outExcel(){
    var table=document.all.cost;
    row=table.rows.length;
    column=table.rows(1).cells.length;
    var excelapp=new ActiveXObject("Excel.Application");
    excelapp.visible=true;
    objBook=excelapp.Workbooks.Add();                              //添加新的工作簿
    var objSheet = objBook.ActiveSheet;
    title=objSheet.Range("D1").MergeArea;                          //合并单元格
    title.Cells(1,0).Value =paytitle.innerHTML.replace(" ","话费单查询"); //输出标题
    title.Cells(1,1).Font.Size =16;
    for(i=1;i<row+1;i++){
    for(j=0;j<column;j++){
    objSheet.Cells(i+1,j+1).value=table.rows(i-1).cells(j).innerHTML.replace(" ","话费单查询");
    }
    }
      objBook.SaveAs("telcost.xls");
    objSheet.Printout;                                             //自动打印
    excelapp.UserControl = true;
}
</script>
```

（4）通过单击“打印”超链接调用自定义 JavaScript 函数 outExcel()，代码如下：

```
<a href="#" onClick="outExcel();">打印</a>
```

运行本示例，在页面中将显示话费列表，单击“打印”超链接，即可将话费列表和自定义的标题信息导出到 Excel 中，并保存到 Excel 的默认工作目录中，同时自动调用打印机进行打印。运行结果如图 16.11 和图 16.12 所示。

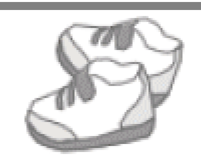




图 16.11　Web 页面中的数据显示

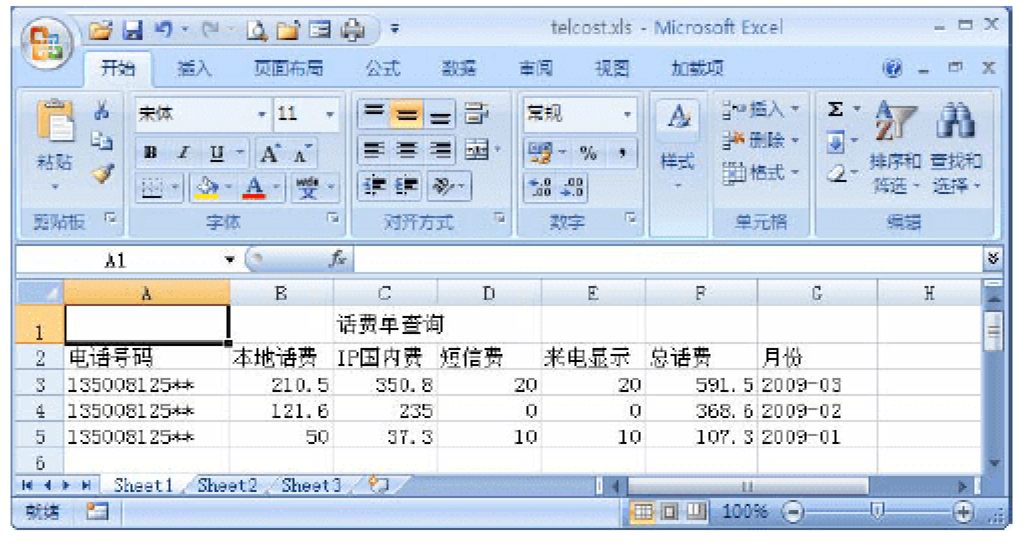


图 16.12　导出到 Excel 中的数据

专家点评

在 Excel 中查看并修改默认文档保存路径的方法如下：选择“工具”→“选项”命令，在弹出的对话框中选择“常规”选项卡，在该选项卡的“默认工作目录”文本中将显示默认的工作目录，读者可以将其修改为其他可用路径。

问题 297　如何将查询结果保存到 Word 中？

问题阐述

在一些基于网络的管理系统中，经常需要将网页中的数据保存到 Word 文档中。那么，如何将查询结果保存到 Word 中呢？

专家解答

将查询结果保存到 Word 文档中，封装自定义的 Word 类，在 Word 类中通过输出缓冲函数将查询结果存储到 Word 中。在 Word 类中所应用的输出缓冲函数包括：

（1）ob_start()：开始输出缓冲。此时 PHP 停止输出，而这以后的输出都被转到一个内部的缓冲中。

（2）ob_get_contents()：该函数返回内部缓冲的内容。这就等于把这些输出都变成了字符串。

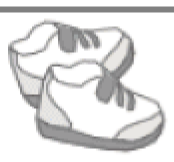
（3）ob_get_length()：返回内部缓冲的长度。

（4）ob_end_flush()：结束输出缓冲，并输出缓冲中的内容。从这以后的输出都是正常输出。

（5）ob_end_clean()：结束输出缓冲，并扔掉缓冲中的内容。

本示例将数据库中的数据保存到 Word 文档中主要是通过自定义类 Word 实现的。Word 类的关键代码如下：

```
<?php
class word
{
    function start()
     {
         ob_start();
     }
     function save($path)
       {
```




```
                $data = ob_get_contents();
                ob_end_clean();
                $this->wirtetoword($path,$data);
            }
        function wirtetoword ($fn,$data)
            {
                $fp=fopen($fn,"wb");
                fwrite($fp,$data);
                fclose($fp);
            }
    }
    ?>
```

在该类中定义了 3 个成员函数，它们实现的功能如下：

（1）start()：该成员函数的作用是定制要保存数据的开始。

（2）save()：该成员函数的作用是定制要保存数据的结束，同时执行将数据库中的数据保存到 Word 中的操作，也就是说，所要保存的数据必须限定在该类的成员函数 start()和成员函数 save()之间。

（3）wirtetoword()：该成员函数的作用是实现将数据以二进制的形式保存到 Word 中。

应用示例

将学生成绩管理系统中查询出的学生成绩数据存储到 Word 文档中。其具体实现步骤如下：

（1）建立 conn.php 文件，实现与数据库的连接。

（2）建立 word.php 文件，在该文件中编写用于将查询结果保存到 Word 中的 Word 类。

（3）建立 index.php 文件，输出查询结果，并将查询结果保存到 Word 中。

本示例的运行结果如图 16.13 和图 16.14 所示，其中图 16.13 所示为所查询到的学生成绩的相关信息，图 16.14 所示为将查询结果保存到 Word 中的结果。



图 16.13　查询结果

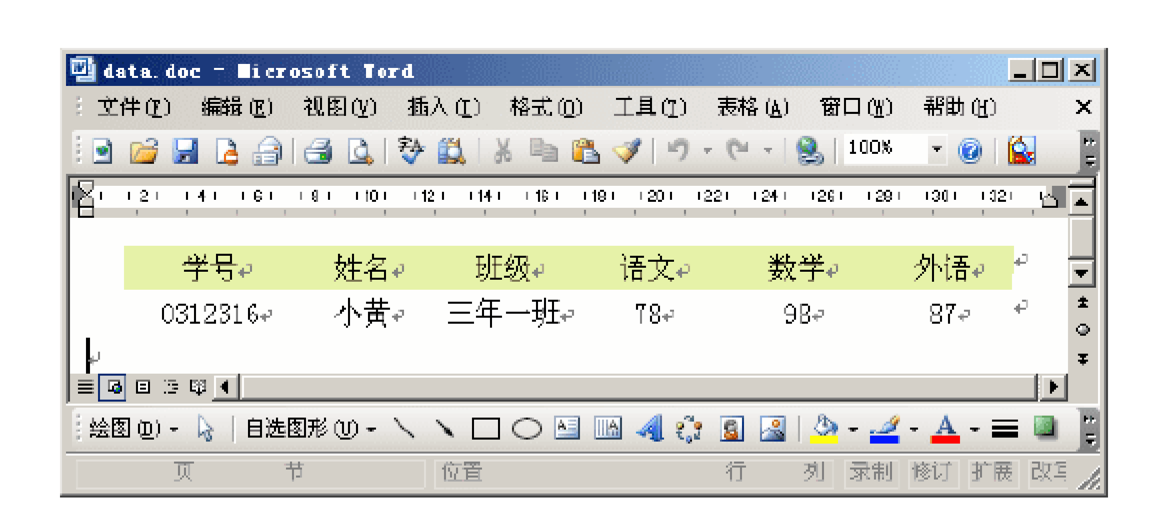
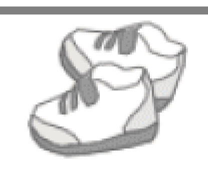


图 16.14　将查询结果保存到 Word 中

问题 298　如何将查询结果保存到 Excel 中？

问题阐述

Excel 在办公系统中被广泛应用，利用该办公软件可以保存大量的统计信息，以方便相关工作人员的打印维护等操作，而大多数管理系统的数据信息都保存到数据库中。那么如何将数据库中的信息保存到 Excel 表格中呢？



专家解答

将数据库中的信息保存到 Excel 表格中，其关键是自定义的 excel 类的封装。在该类中定义 3 个成员函数，其中 start()用于限定要保存的数据的开始；save()用于限定要保存的数据的结束，并调用成员函数 wirtetoexcel()实现将查询结果以二进制的形式保存到 Excel 表格中。excel 类的语法如下：

```
<?php
class excel{
    function start(){
        ob_start();
      }
     function save($path){
          $data = ob_get_contents();
          ob_end_clean();
          $this->wirtetoexcel($path,$data);
      }
    function wirtetoexcel ($fn,$data){
         $fp=fopen($fn,"wb");
         fwrite($fp,$data);
         fclose($fp);
     }
}
?>
```

应用示例

将从数据库中查询出的数据保存到 Excel 表格中。其具体实现步骤如下：

（1）建立数据连接文件 conn.php 实现与数据库的连接。

（2）封装 excel 类。

（3）创建 index.php 文件，实现将查询结果保存到数据库中。

运行本示例，在图 16.15 所示的表单中输入要查询的学生的学号，单击“将表格内容保存到 Excel”按钮，将查询结果保存到 Excel 表格中，如图 16.16 所示。



图 16.15　查询结果

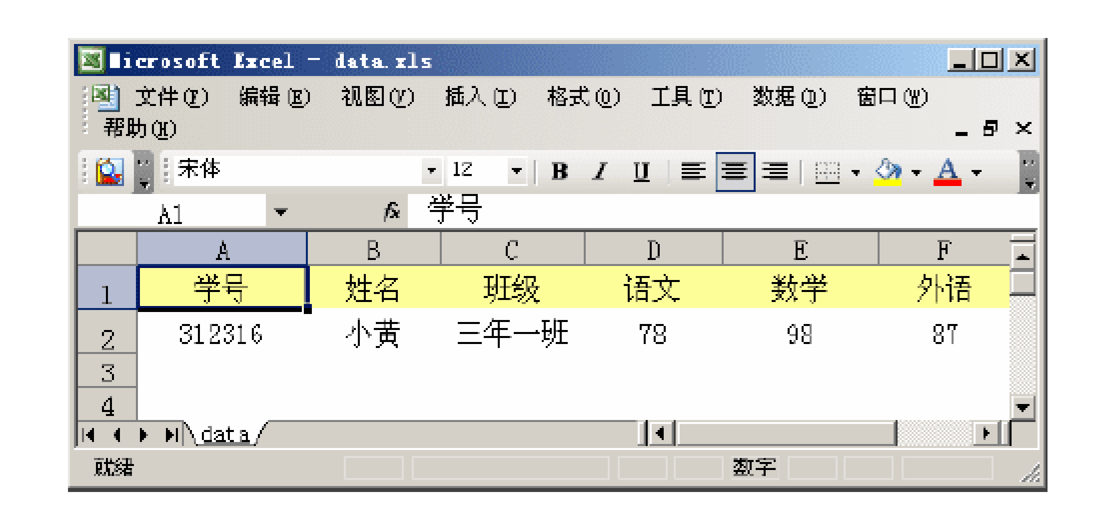
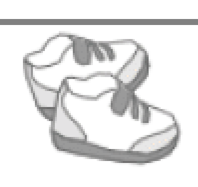


图 16.16　将查询结果保存到 Excel 中的结果

问题 299　如何实现在线支付（工商银行）？

问题阐述

随着 Internet 的不断发展，电子商务已经成为 21 世纪网络发展的主流，网上购物和网上支



付已经成为一种时尚。那么，如何与工商银行进行交互，完成在线支付操作呢？

专家解答



1. 解决思路

要在电子商务网站中通过工商银行完成在线支付的操作，其具体的开发流程如图 16.17 所示。

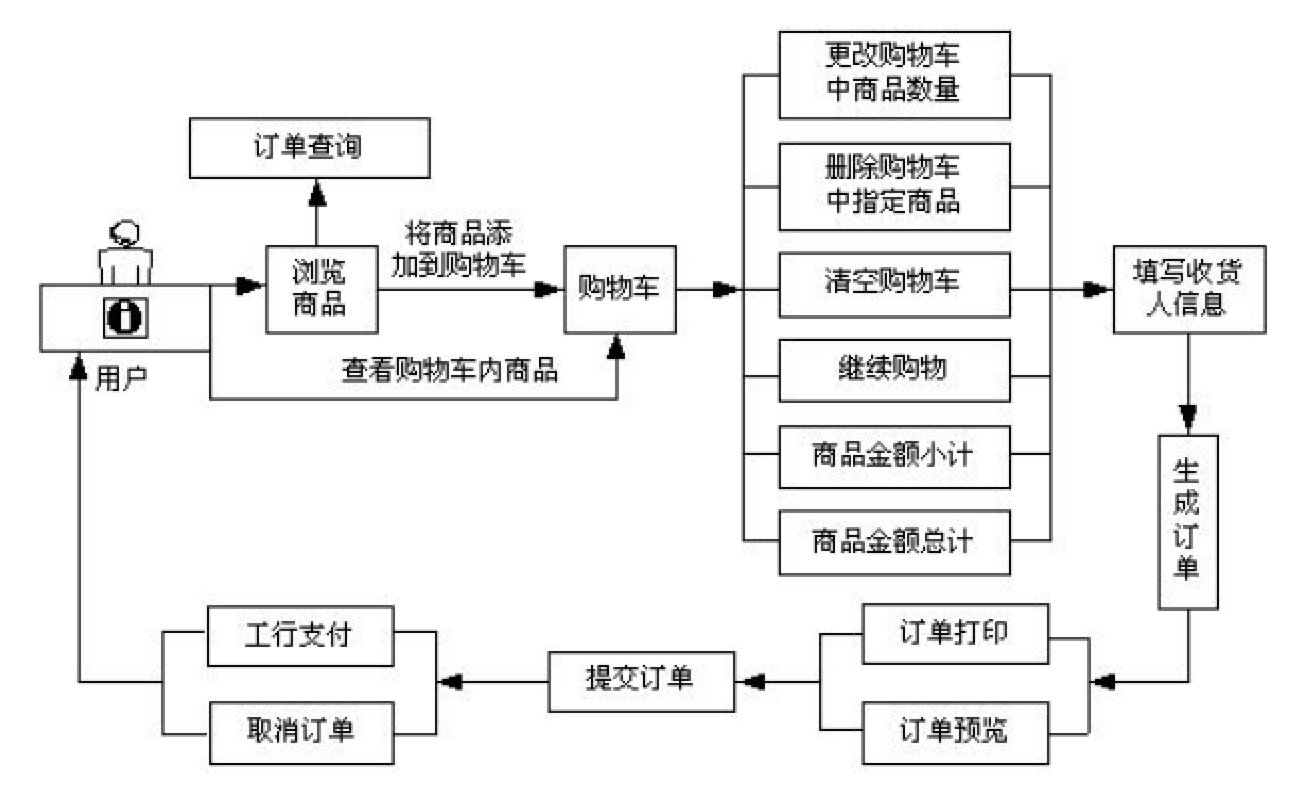


图 16.17　在线支付的操作流程

从图 16.17 中可以看出，工商银行在线支付是整个电子商务流程中最重要的一个操作环节，在执行该操作之前，必须逐一完成前面的操作环节，否则单独执行工商银行在线支付没有任何意义。所以，在开发本示例时，不只要完成工商银行在线支付功能的开发，同时还要进行商品浏览、购物车、生成订单和提交订单等功能的开发。

2. 具体方法

工商银行在线支付根据工商银行提供的在线支付接口，将订单中的数据以表单的形式提交到工商银行指定的网址中。具体详细的参数设置可以参考工商银行提供的文档说明书和示例程序。

在实现与工商银行接口的连接之前，首先要从工商银行提供的说明书中找到下面两个文件：

（1）开发 API 接口\LIB\windows\win32\infosecapi.dll（Windows XP 系统使用的文件）。

（2）开发 API 接口\ASP\ICBCEBankUtil.dll。

将其复制到系统文件夹下的\WINDOWS\system32 下，在运行命令中输入“regsvr32 ICBCEBankUtil.dll”，加载服务。

通过 PHP 实现网站与工商银行接口交互的方法如下：

首先，通过 com 类中的 init()方法对订单签名数据和商城的公钥进行加密处理，并且将指定的数据以表单的形式提交到工行指定的页面中。其关键代码如下：

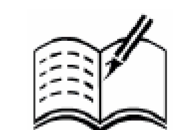
```
        $src="ICBC_PERBANK_B2C1.0.0.0".$infomer["merid"].$infomer["meracct"].$returnaddress."HS"
.$ddnumber.$amount."0010".$nowtime."0";
        $com=new com('ICBCEBANKUTIL.B2CUtil');
        $rc=$com->init("c:\WINDOWS\user.crt","d: \user.crt","d: \user.key","00");
        $ssrc=$com->signC($src, strlen($src));                    //订单签名数据
        $rc=$com->verifySignC($src,strlen($src), $ssrc, strlen($ssrc));
        $cert=$com->getCert(1);                                   //商城证书公钥
```

完成订单签名数据和商城证书公钥的加密处理，这里将商户的测试证书存储到 C:\WINDOWS 目录下，设置商户的私钥口令是 00。

然后，创建 form 表单，将指定的数据提交到工行指定的网站中。关键代码如下：







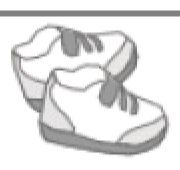
```
    <form   action="https://mybank.icbc.com.cn " method="post" name="form_bank">
     <input name="interfaceName" type="hidden" value="ICBC_PERBANK_B2C"/>
     <input name="interfaceVersion" type="hidden" value="1.0.0.0"/>
        <input name="orderid" type="hidden" value="<{$ddnumber}>"/>
        <input name="amount" type="hidden" value="<{$amount}>"/>
     <input name="curType" type="hidden" value="001"/>
        <input name="merID" type="hidden" value="<{$merid}>"/>
        <input name="merAcct" type="hidden" value="<{$meracct}>"/>
     <input name="verifyJoinFlag" type="hidden" value="0"/>
     <input name="notifyType" type="hidden" value="HS"/>
       <input name="merURL" type="hidden" value="<{$returnaddress}>"/>
     <input name="resultType" type="hidden"   value="0"/>
        <input name="orderDate" type="hidden" value="<{$nowtime}>"/>
     <input name="merSignMsg" type="hidden" value="<{$ssrc}>" />
     <input name="merCert" type="hidden" value="<{$cert}>" />
 </form>
```

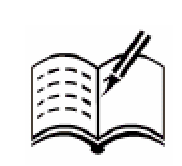
3. 应用示例

吉林省明日科技有限公司的编程词典网（http://www.mrbccd.com）就是一个电子商务网站，它完美地实现了编程词典的在线购买功能，并且为用户提供了多种支付方式，如图 16.18 所示。



图 16.18　编程词典网站的支付方式





在本示例中，将对编程词典网站中的工商银行在线支付功能进行剖析。其具体实现步骤如下：

（1）本示例应用 Smarty 进行开发，通过 ADODB 操作 MySQL 数据库，并将 Smarty 模板和 ADODB 类库存储在 system 文件夹下。同时将连接、操作 MySQL 的方法以及 Smarty 的配置方法都定义到类文件中，并存储在 system 文件夹下。system 文件夹的架构如图 16.19 所示。

（2）创建动态 PHP 文件 index.php 和模板文件 index.phtml，实现商品的分页、分类浏览，并且创建购买和查看购物车的超链接。

在 index.php 中，调用包含文件，完成数据库的连接，应用数据库管理类 AdminDB()中的 ExecSQL()方法读取数据中的数据，并将查询结果赋给模板变量。

应用分页类 SepPage()中的 ShowDate()方法，执行分页查询，实现分页功能，将分页结果赋给模板变量；应用分页类 SepPage()中的 ShowPage()方法执行分页操作，完成分页超链接的创建，并将返回结果赋给模板变量。最终指定模板页 index.phtml。

在 index.phtml 模板页中，根据 PHP 模板变量传递的值，应用 section 循环语句完成商品信息的分页输出，同时创建购买商品和查看购物车的超链接，传递的参数为商品的 ID。

（3）创建购物车。在购物车功能模块中包含更改商品数量、删除商品、商品小计、商品总计、清空购物车和结算等。购物车的操作页面如图 16.20 所示。

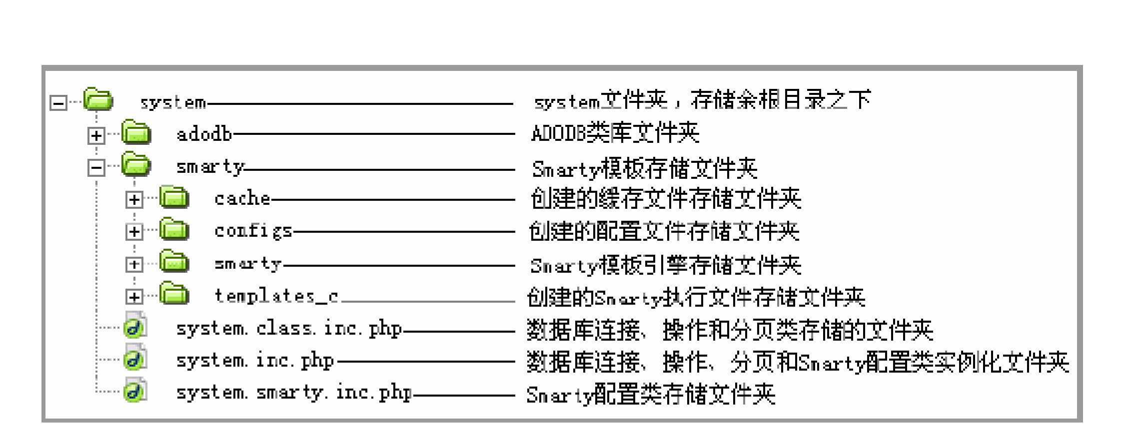


图 16.19　system 文件夹的架构



图 16.20　购物车操作页面

（4）创建、生成订单，完成订单打印和打印预览功能的实现。创建订单应用的是 shopping_order_form.php 和 shopping_order_form.phtml 文件，订单处理应用的是 shopping_order_form_ok.php 文件。

生成订单通过 create_order_form.php 和 create_order_form.phtml 文件来完成。

订单打印和打印预览功能应用的是 printwindow.php 和 printwindow.phtml 文件。首先在 create_order_form.phtml 文件中应用 openprintwindow()函数将参数传递到 printwindow.php 文件中，然后在该文件中完成订单打印和预览的操作。

（5）创建 submit_order_form.php 和 submit_order_form.phtml 文件，实现订单处理。

在 submit_order_form.php 中根据订单号查询出订单的详细信息，并将查询结果赋给指定的模板变量；实现网站与工商银行接口的连接，定义模板变量将经过加密的数据传递到模板页的表单中，通过 form 表单向工商银行指定的网站中传递数据。其关键代码如下：

```
        //组织要签名的数据串
        $src="ICBC_PERBANK_B2C1.0.0.0".$infomer["merid"].$infomer["meracct"].$returnaddress."HS"
.$ddnumber.$amount."0010".$nowtime."0";
        $com=new com('ICBCEBANKUTIL.B2CUtil');
        $rc=$com->init("c:\WINDOWS\user.crt","c:\WINDOWS\user.crt","c:\WINDOWS\user.key","00");
        $ssrc=$com->signC($src, strlen($src));             //订单签名数据
        $rc=$com->verifySignC($src,strlen($src), $ssrc, strlen($ssrc));
        $cert=$com->getCert(1);                            //商城证书公钥
        $smarty->assign("ssrc",$ssrc);                     //定义模板变量传递加密的订单签名数据
```






```
        $smarty->assign("cert",$cert);                    //定义模板变量传递加密的商城公钥
        $nowtime=date("YmdHis");
        //将指定的数据通过模板变量传递到表单中
        $sqlmer=mysql_query("select merid,meracct from tb_system where mark=1",$conn);
        $infomer=mysql_fetch_array($sqlmer);
        $returnaddress="http://www.cccxy.com/getinfo.asp";
        $smarty->assign("ddnumber",$ddnumber);
        $smarty->assign("nowtime",$nowtime);
        $smarty->assign("returnaddress",$returnaddress);
        $smarty->assign("merid",$infomer["merid"]);
        $smarty->assign("meracct",$infomer["meracct"]);
        $smarty->assign("src",$src);
```

在 submit_order_form.phtml 文件中创建 form 表单，将从模板变量中获取的数据通过表单提交到工商银行指定的网站中。其关键代码如下：

```
        <!--工行支付的接口操作，提交的数据-->
        <form   action="https://mybank.icbc.com.cn/" method="post" name="form_bank">
            <input name="interfaceName" type="hidden" value="ICBC_PERBANK_B2C"/>
            <input name="interfaceVersion" type="hidden" value="1.0.0.0"/>
                <input name="orderid" type="hidden" value="<{$ddnumber}>"/>
                <input name="amount" type="hidden" value="<{$amount}>"/>
            <input name="curType" type="hidden" value="001"/>
                <input name="merID" type="hidden" value="<{$merid}>"/>
                <input name="merAcct" type="hidden" value="<{$meracct}>"/>
            <input name="verifyJoinFlag" type="hidden" value="0"/>
            <input name="notifyType" type="hidden" value="HS"/>
              <input name="merURL" type="hidden" value="<{$returnaddress}>"/>
            <input name="resultType" type="hidden"   value="0"/>
                <input name="orderDate" type="hidden" value="<{$nowtime}>"/>
            <input name="merSignMsg" type="hidden" value="<{$ssrc}>" />
            <input name="merCert" type="hidden" value="<{$cert}>" />
        </form>
        <td width="124"><img src="images/sy_18.jpg" width="70" height="25" onClick="javascript:form_
bank.submit();" style="cursor:hand"/></td>
        <!--————————————————————-->
```

这里给出编程词典网站通过工商银行进行在线支付的操作页面，如图 16.21 所示。

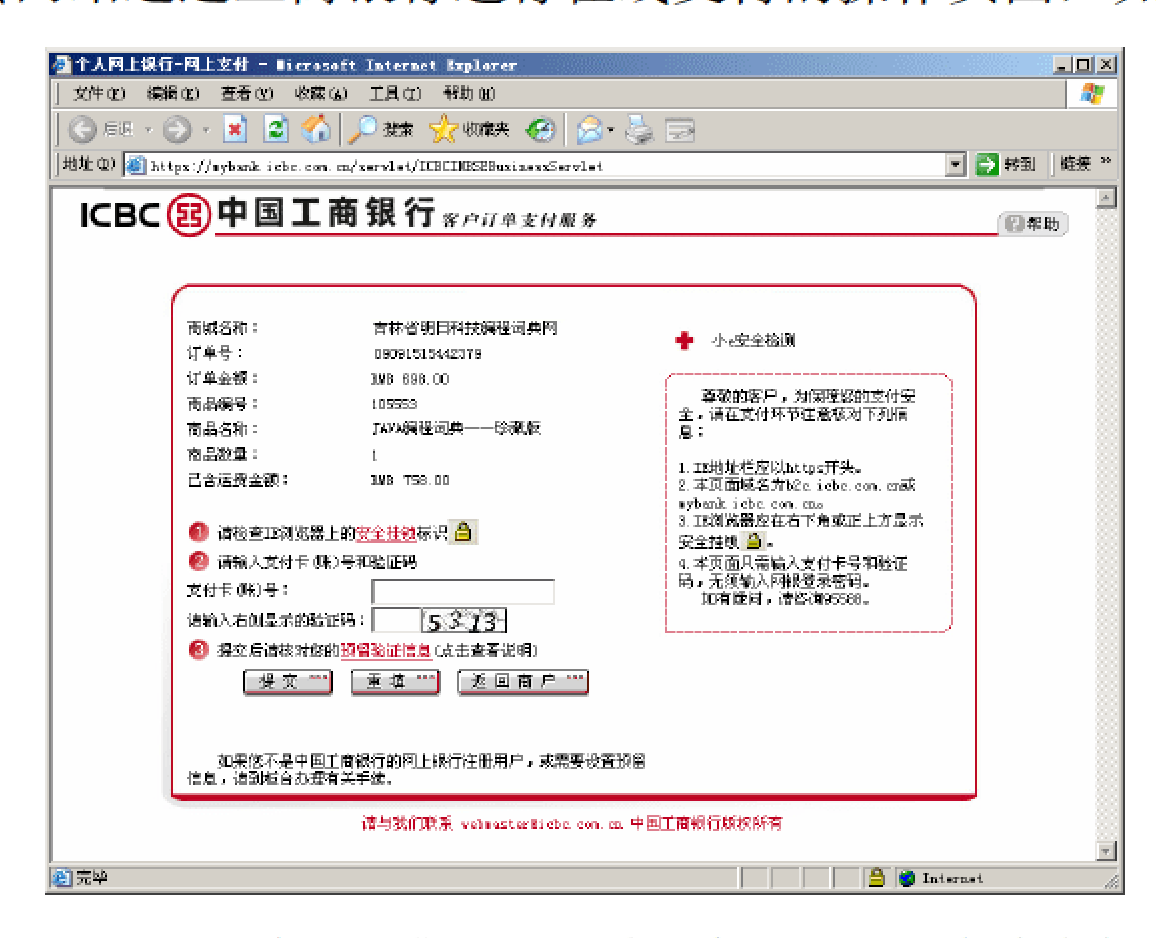
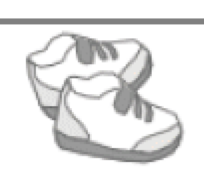
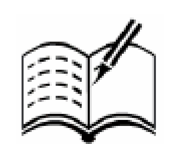


图 16.21　编程词典网站通过工商银行进行在线支付





Note

专家点评

这里讲解的是在通过 PHP 语言开发的电子商务网站中如何与工商银行进行交互的方法，其中涉及工商银行提供的参数和设置都是个人虚拟的，真实的参数将在实际的运作中由工商银行提供。当实际需要使用此项功能时，需向工商银行进行申请，工商银行会提供相应的参数。

问题 300　如何实现在线支付（支付宝支付）？

问题阐述

通过工商银行可以实现在线支付的操作，而通过支付宝也可以实现在线支付的操作。下面我们就来探讨一下如何通过支付宝实现在线支付。

专家解答

使用支付宝进行在线支付，必须先在支付宝中注册一个支付宝账户，然后下载支付宝提供的接口文件，通过该接口文件可实现电子商务网站与支付宝的交互。

在电子商务网站中需要嵌入几个支付宝提供的文件，如图 16.22 所示。

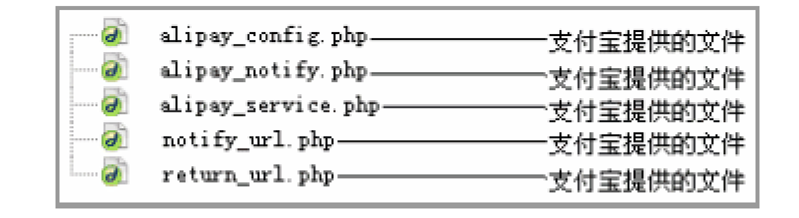


图 16.22　在示例根目录下嵌入的支付宝提供的文件

在支付宝提供的接口文件中提供了完整的示例程序，只要依照示例程序中的操作方法，将电子商务网站中指定的数据提交到支付宝提供的数组参数中，支付宝即会通过自己定义的方法将数据提交到指定的网站中。实现网站与支付宝交互的关键代码如下：

```
require_once("alipay_service.php");                    //调用支付宝提供的文件
require_once("alipay_config.php");                     //调用支付宝提供的文件
$array=explode("@",$info["spc"]);
$arraynum=explode("@",$info["slc"]);
$arrayinfo=array();                                    //创建数组
$arrayinfo_count=array();                              //创建数组
for($i=0;$i<count($array);$i++){
    if($array[$i]!=""){
        $m=$i+1;
        $sqlcart=mysql_query("select * from tb_commodity where tb_commodity_id='".$array[$i]."'",
$conn);
        $infocart=mysql_fetch_array($sqlcart);                 //读取数据库中数据
        array_push($arrayinfo,$infocart[tb_commodity_name]);//将购物商品的名称写入数组中
        array_push($arrayinfo_count," 商 品  $m  ： $infocart[tb_commodity_name]  数 量 ：
$arraynum[$i]");                                               //将购物商品的数量写入数组中
    }
}
$commodity_name=implode(',',$arrayinfo);                       //获取商品名称
$commodity_count=implode(',',$arrayinfo_count);                //获取商品描述信息
$parameter = array(
"service" => "trade_create_by_buyer",     //交易类型，必填实物交易=trade_create_by_buyer
"partner" =>$partner,                     //合作商户号
```





```
"return_url" =>$return_url,            //同步返回
"notify_url" =>$notify_url,            //异步返回
"_input_charset" => $_input_charset,   //字符集，默认为 GBK
"subject" => $commodity_name,          //商品名称，必填
"body" => $commodity_count,            //商品描述，必填
"out_trade_no" => $ddnumber,           //商品外部交易号，必填，每次测试都必须修改
"logistics_fee"=>$yprice,              //物流配送费用
"logistics_payment"=>'BUYER_PAY',      //物流配送费用付款方式：BUYER_PAY（买家支付）
"logistics_type"=>'EXPRESS',           //物流配送方式：EMS(EMS)、EXPRESS（其他快递）
"price" => $amount,                    //商品单价，必填
"payment_type"=>"1",                   //默认为 1，不需要修改
"quantity" => "1",                     //商品数量，必填
"show_url" => $show_url,               //商品相关网站
"seller_email" => $seller_email        //卖家邮箱，必填
);
$alipay = new alipay_service($parameter,$security_code,$sign_type);
$link=$alipay->create_url();
$smarty->assign("link",$link);
```

专家点评

这里讲解的是通过 PHP 来实现电子商务网站与支付宝进行交互的方法，其中涉及支付宝提供的参数和设置都是虚拟的，真实的参数将在实际的运作中由支付宝提供。